

---

## INDEX

---

The purpose of this index is to help you to search for information in this book. After each term mentioned in the index, there are page numbers that help you to locate the pages where the term is discussed. Each page number indicates a page on which the discussion of a term begins. In some cases, the discussion continues on the following pages.

The file names of the example programs (e.g. **First.java**, **Sum.java**, **Game.java**, etc.) are not mentioned in this index. To locate the example programs in this book, please use the alphabetical list on the introductory pages of this book.

Terms related to the imaginary computer (Chapter 4) are not widely listed in this index. Because the imaginary computer is a special subject that is discussed only in a single chapter, it should be possible to manage that information by using the table of contents.

### Symbols

! NOT operator 146  
 != not equal operator 135  
 " double quote character 102, 121, 203, 555  
 \$ in format specifiers 125  
 % in format specifiers 125  
 % remainder operator 138, 265, 280  
 %s format specifier for strings 506  
 & bitwise AND operator 531  
 && logical AND operator 146  
 && operator vs. & operator 552  
 ( opening parenthesis 134  
 () empty parentheses 108, 244, 252  
 ) closing parenthesis 134  
 \* multiplication operator 113  
 \*/ terminates a comment 313  
 \*= multiply and assign operator 559  
 + addition operator 113, 121  
 + concatenation operator 121  
 ++ increment operator 155, 157  
 += add and assign operator 559  
 - subtraction operator 113  
 -- decrement operator 157  
 -= subtract and assign operator 559  
 ... means "some program lines" 12, 274  
 ... specifies zero or more parameters 274  
 .bak, name extension for backup files 466  
 .class, name extension for executable Java files 10  
 .exe files vs. .class files 28  
 .java, name extension for Java source files 10  
 .jpg files 497  
 .txt, name extension for text files 452  
 / division operator 113

/\* begins a comment 313  
 // double slash, begins a comment line 17, 18, 73, 313  
 ; semicolon 14, 18  
 < less than operator 135  
 << left shift operator 531  
 <= less than or equal operator 135  
 = basic assignment operator 113  
 == equal operator 135  
 > greater than operator 135  
 >= greater than or equal operator 135  
 >> right shift operator 531  
 >>> zero fill right shift operator 531  
 [] empty (square) brackets 178, 188  
 \ backslash character 203, 555  
 \0 NULL character 555  
 \b backspace character 203, 555  
 \n newline character 102, 203, 225, 555  
 \r carriage return character 203, 225, 453, 550, 555  
 \t tabulator character 203, 225, 555  
 ^ exclusive OR operator 531  
 \_ underscore character 108  
 | bitwise OR operator 531  
 || logical OR operator 146  
 || operator vs. | operator 552  
 ~ complement operator 531  
 ` single quote character 141, 203, 555

### Numerics

0x, prefix to form hexadecimal literals 117  
 1\$, 2\$, etc. in format specifiers 125  
 10-finger typing system vi  
 24-hour time format 423

### A

a.m./p.m. time format 423  
 abbreviated names 17  
 abstract classes 422, 424, 520  
 abstract classes vs. final classes 429  
 abstract keyword 424, 567  
 abstract methods 422  
 access modifiers 298, 398  
     protected keyword 416  
 accessing files in Java programs 451  
 accessor methods 313, 339, 343, 351, 365  
 AccountWithCredit, an example class 418  
 acres, unit of area 319  
 action statements 99, 100  
 action statements vs. declaration statements 14  
 actual parameters 250, 268  
 add and assign operator += 559  
 add(), ArrayList method 465, 502, 507, 517, 518  
 add(), Calendar method 524  
 add\_new\_collection\_item() method 491  
 add\_new\_item(), Collection method 485  
 addAll() method of class Collections 509  
 addAll(), ArrayList method 505, 518  
 addition operator + 113, 121

- additive arithmetic operators 559
  - addresses, numerical 51
  - advice for studying v
  - age, calculation of 347, 523
  - algorithms, sorting 262
  - allocation of memory 188, 190, 272
  - alphabetical comparison of strings 220
  - American date format MM/DD/YYYY 342, 388
  - AmericanTime, an example class 425
  - Animal, an example class 308, 362
  - API (Application Programming Interface) 431
  - Apollo 11 landing 344
  - append(), StringBuilder method 229
  - appending data to the end of a file 472
  - appending text to the end of a file 455
  - applications 9
  - Arabic numbers 32, 33
  - argument lists of variable length 274
  - arguments vs. parameters 250
  - arguments, see "parameters" 244
  - arithmetic expressions 138, 558
    - in array references 179
  - arithmetic operators 113, 138, 280
  - ArrayList class 502–512, 518
    - add() method 465, 502, 507, 517, 518
    - addAll() method 505, 518
    - get() method 507
    - set() method 507
    - size() method 503
    - type parameters 505
  - arrays
    - array creation statement 189
    - array declaration statement 189
    - arrays of integers 178–185
    - arrays of objects 557
    - arrays of strings 231, 234
    - as method parameters 257
    - declaring arrays 178, 556
    - default values 192
    - "foreach" loops process arrays 176
    - initialized array of strings 231
    - initialized arrays 191
    - jagged arrays 196
    - length of an array 183
    - multidimensional arrays 196, 406
    - ragged arrays 196
    - rectangular arrays 196
    - referring to array elements 179
  - Arrow Down key 26, 573
  - Arrow Up key 26, 573
  - ASCII coding system 45, 105
  - ask\_numbers\_to\_array() method 260, 265
  - assembly languages are difficult 94
  - assignment operator = 113, 115
  - assignment statements 113, 115, 558
    - left-side expressions 558
  - associativity of operators 112, 121
  - associativity of the dot operator . 323
  - astronomical facts 342
  - autoboxing 432, 502
  - automatic calls 385, 400
  - automatic memory management 301, 325
  - automatically generated constructors 400
  - auxiliary memory devices 5, 50
  - available() method 489
- ## B
- Babbage, Charles 205
  - backslash character \ 203, 555
  - backspace character \b 203, 555
  - backup (.bak) files 466
  - backup copies of programs 577
  - balloons of text viii
  - BankAccount, an example class 296
  - BankAccount, another version of the class 302
  - BankAccount, third version of the class 416
  - base class 384
  - base-10 system, base-16 system, etc.
    - see "numbering systems"
  - BetterDate, an example class 386
  - big-endian order of bytes 106
  - BilingualTranslation, an example class 510
  - binary files vs. text files 469
  - binary literals do not exist 530
  - binary numbering system 33
  - binary numbers 42
    - printing integers in binary form 536
  - binary-to-decimal conversion 36
  - binding, dynamic (late) vs. static (early) 427
  - birthdays, day of week 347, 523
  - birthdays, planning celebrations 347, 523
  - bit operators 530–537
  - bit, binary digit 40
  - bitwise AND operator & 531
  - bitwise logical operators 552
  - bitwise OR operator | 531
  - blocks of statements 167, 171, 276
  - blocks, nested program blocks 276
  - body of a loop 159
  - body of method 259
  - Boole, George 134
  - boolean expressions 134, 140, 470, 558
  - boolean keyword 567
  - boolean literals (true, false) 174
  - boolean methods 490
  - boolean type 105, 175
  - Boolean, standard wrapper class 433
  - boxing 432, 502
  - boxing vs. unboxing 436, 437
  - braces { }, rules for using 167, 571
  - brackets [] 188
  - break keyword 567
  - break statements 147, 150
  - BufferedReader class 454, 458
    - readLine() method 459
  - bug can mean any small insect 96
  - byte keyword 567

- byte type 105
- Byte, standard wrapper class 433
- bytecode instructions 268, 270
  
- C**
  
- C programming language 272, 452
- C# programming language 112, 452
- C++ programming language 112, 305, 452
- Caesar, Julius 237, 347
- calculate\_mean\_value() method 261
- Calendar class 390, 521
- Calendar constants 524, 525
- calendar reform 347
- callee, a method that is called 242
- caller, a method that calls 242
- calling methods 240, 513
  - stack usage 268–273
- calling superclass version of a method 419
- calories, unit of energy 319
- capacity(), StringBuilder method 226
- capitalization of names iv, 252
- carriage return character `\r` 203, 225, 453, 555
- case keyword 147, 567
- case sensitivity 22
- casting 565
- catch keyword 171, 567
- celebrations, planning 347, 523
- centimeters, unit of length 319
- change\_text(), TextWindow method 410
- char keyword 567
- char type 105
- character codes 141, 160, 200, 452, 453, 532
  - whitespace characters 225
- character codes, hexadecimal output 206
- character coding 45, 46
- character literals 141, 555
- Character, standard wrapper class 433
- charAt() String method 204, 205, 220
- checked exceptions 442, 443
- chronological distance between dates 345, 357
- Churchill, Winston 521
- .class file, executable Java file 10
- .class files vs. .exe files 28
- class keyword 567
- class methods vs. instance methods 208
- class variables 397
- classes
  - abstract classes 422, 424
  - access modifiers 398
  - accessor methods 339, 343, 351, 365
  - base class 384
  - calling another instance method 352
  - class declarations 566
  - class hierarchy 404
  - class scope 277
  - constructor methods 300
  - copy constructors 306
  - default constructors 307
  - fields of classes 290
  - final classes 429
  - graphical class descriptions 365
  - hierarchies of classes 401, 405, 415
  - inheritance between classes 384
  - initialization of data fields 316
  - polymorphic methods 414, 510
  - several constructors in a class 306
  - subclasses and immediate subclasses 400
  - superclass 384
  - superclasses and immediate superclasses 400
- classwide data 277
- classwide variables 275
- clock signal 54
- clone() method of class Object 434
- close(), method to close files 455, 457, 466
- coding of textual information 45, 46, 452
- Collection, an example class 484
- CollectionItem, an example class 482
- CollectionMaintainer, an example class 490
- Collections.addAll() method 509
- Collections.sort() method 518
- Columbus, Christopher 344, 347
- command line
  - command line parameters 467
  - directing screen output to file 399
- command prompt window, opening and adjusting 19
- command prompt window, working with 26
- command\_line\_parameters, parameter of main() 278
- commands, MS-DOS operating system 19
- comments
  - comments inside `/*` and `*/` 313
  - double slash `//` comments 17, 73, 313
  - text ignored by the compiler 17
  - using comments in program testing 575
- Comparable interface 514, 515, 516
- compareTo() method of class Event 516
- compareTo() method of Comparable 515
- compareTo() String method 212, 220
- compareToIgnoreCase() String method 220
- comparison operators 135
- compilation 8, 350
  - compiling Java programs 25, 26
  - compiling with JCreator 29
  - process of compilation studied 130
  - recompiling programs 26
  - several .java files in compilation 344
- compilation errors, correcting 573
- compiler generated calls 385
- compiler, installation of 20
- complement operator `~` 531
- complementing truth values 145
- compound interest 194
- computer system 9
- concat() String method 221
- concatenating String objects 214
- concatenation operator `+` 121
- conditional logical operators 552
- console window 19
- const keyword 567

constants 370  
 constants, literal constants 113  
 constructors 300, 306
 

- automatic calling of superclass constructor 400
- constructor calling another constructor 363
- constructors and inheritance 385, 386
- copy constructors 306
- default constructors 307, 385, 400
- implicit calling of constructors 385
- initialization of data fields of a class 316

 constructors of class `String` 221  
`contains()` `String` method 213, 221, 320, 377, 462  
`contentEquals()` `String` method 221  
 contents of this book x  
 continue keyword 567  
 continuing a statement on following line 109  
 control structures 560, 561  
 Control-C character 575  
 conventions for writing hexadecimal numbers 39  
 conventions for writing method names 252  
 Conversion, an example class 318  
 conversions
 

- explicit type conversions 161, 164, 267
- summary of some mechanisms 565

 conversions, explicit 532  
`convert()`, Conversion method 318, 320  
`convert_string_uppercase()` method 535  
 copy constructors 306  
 copying objects 434  
`copyValueOf()` `String` method 221  
 CR, carriage return character 453  
 creating arrays 189  
 creation of objects 300  
`CurrentDate`, an example class 390  
`CurrentTime`, an example class 424  
`currentTimeMillis()`, method of class `System` 544

## D

Darwin, Charles 521  
 data encapsulation 339  
`DataInputStream` class 472, 473, 482, 489  
`DataOutputStream` class 471, 472, 473, 476, 488  
 date formats 342, 388  
`Date`, an example class 350–360, 384  
`DateDistance`, an example class 361  
`day()`, `Date` method 351  
 deallocation of memory 272  
 decimal numbering system 101  
 decimal numbers 101  
 decimal-to-binary conversion 37  
 decisions, see "if constructs", "switch-case constructs"  
 declaration statements vs. action statements 14  
 declarations
 

- array declarations 178, 189, 556
- class declarations 566
- `String` declarations 200
- variable declarations 100, 556
- vs. executable statements 100

declarators of methods 244, 259  
`DecoratedTextWindow`, an example class 411  
 decrement operator `--` 157  
`decrement()`, `Date` method 356  
 default constructors 307, 385
 

- automatically generated 400

 default keyword 567  
 default package 336  
 default values 192  
 definitions of variables 100  
`delete()`, `StringBuilder` method 229  
`deposit_money()`, `BankAccount` method 296, 302, 416  
 designing programs 332  
 destruction of objects in heap memory 301  
 destructors do not exist in Java 305  
 digits 32  
 dimensions of arrays 196  
 directing screen output to file 399  
 dividing by zero 280  
 division operator `/` 113, 280  
 do keyword 567  
 do-while loops 164–165, 561  
 documentation, Java API 22  
 dollar sign `$` in format specifiers 125  
 dot operator `.`

- associativity of 323

 double keyword 567  
 double quote character `"` 121, 203, 555  
 double slash `//` begins a comment line 313  
 double type 110  
`Double`, standard wrapper class 433  
`Double.parseDouble()` method 565  
 downloading
 

- Java compiler 20
- programs presented in this book 23

 dynamic binding 427  
 dynamic memory allocation 326  
 dynamic strings 226

## E

e-mail 46  
 e.g. means "for example"  
 early binding 427  
 Earth, equatorial radius 110  
 Earth, length of equator 115  
 echoing 88  
 Einstein, Albert 387  
 electronic Java documentation 22  
 elements
 

- referring to array elements 178

 else keyword 567  
`embed_text_in_window()`, `TextWindow` method 410  
 empty (square) brackets `[]` 188  
 empty parentheses `()` 108, 244, 252  
 empty string 204, 467  
`Empty.java` 571  
 encapsulation of data 294, 339, 398  
 endian, little-endian vs. big-endian 106

endless loop 154  
 endsWith() String method 221  
 EnglishCalendar, an example class 394  
 enum keyword 567  
 enums 499, 500  
 equal operator == 135  
 equality of String objects 210, 467  
 equals() method of class Object 434  
 equals() method, how to write it 445  
 equals() String method 210, 212, 222  
 equalsIgnoreCase() String method 222  
 Error class 443  
 errors in programs  
     correcting compilation errors 573  
     typical logical errors 574  
 errors made by users 171  
 escape character, backslash \ 203, 555  
 escape sequence characters 203  
 etc. means "et cetera", "and so on"  
 European date format DD.MM.YYYY 342, 388  
 EuropeanTime, an example class 425  
 evaluation of expressions 112, 113, 179  
 even numbers vs. odd numbers 197  
 Event, an example class 516  
 example programs, downloading 23  
 example programs, list of xiv  
 Exception class 443  
 exceptions 171, 438–444  
     checked exceptions 442  
     exception specifications 442  
     unchecked exceptions 442  
     unhandled exceptions 120  
 exclusive OR operator ^ 531  
 .exe files vs. .class files 28  
 executable programs 8  
 execution of Java programs 25, 28  
 exercises in this book, list of xvii  
 exercises, practical advice for 571  
 exists(), method of class File 455  
 explicit type conversions 161, 164, 267  
 exponential notation 110  
 expressions 558  
     arithmetic expressions 138, 558  
     boolean expressions 140, 558  
     parentheses, meaning in expressions 123  
     relational expressions 141  
 extending an interface 520  
 extends keyword 567

## F

F, suffix for float literals 117  
 factorial, mathematical term 252  
 false, boolean literal 174, 567  
 feed(), Animal method 308  
 feet, unit of length 319  
 fields of classes 290  
 File class  
     exists() method 455

file name extension 5  
 FileInputStream class 473, 489  
 FileNotFoundException 458, 475  
 FileOutputStream class 473, 488  
     write() method 471  
 FileReader class 454, 458, 460  
 files  
     accessing files in Java programs 451  
     close() method 466  
     opening files 451, 456, 472  
     storing objects in file 480  
     storing screen output to file 399  
     summary of file handling 564  
     text files 452  
 FileWriter class 454, 455, 460  
 fill\_with\_character(), Window method 407  
 final classes and methods 429  
 final keyword 370, 429, 498, 567  
 finalize() method 305  
 finalize() method of class Object 434  
 finally blocks 457  
 finally keyword 444, 568  
 finding errors in programs 573  
 findInLine(), Scanner method 169  
 Finland, date of independence 385  
 flip-flops 42  
 float keyword 568  
 float type 110  
 Float, standard wrapper class 433  
 floating-point literals 117, 192, 555  
 floating-point variables 110  
 floppy disk drive 5  
 flowcharts 153  
 flush() method 455  
 for keyword 568  
 for loops 158–163, 276, 561  
 "foreach" loops 176, 217, 508, 517  
 formal parameters 250, 268  
 format specifiers 116, 128, 523, 526  
     %s format specifier for strings 506  
 format() String method 126, 222, 274, 360, 369  
 Formatter class 128  
 formatting screen output 125–128  
 FrameWindow, an example class 409

## G

gallons, unit of volume 319  
 garbage collection 209, 435  
 garbage collector 301, 325  
 generic classes and types 502  
 generic interfaces 514  
 Germany, date of reunification 385  
 get(), ArrayList method 507  
 get(), Calendar method 524  
 get\_brief\_president\_info(), President method 369  
 get\_date\_print\_format(), Date method 351  
 get\_day\_of\_week(), Date method 355  
 get\_distance\_to(), Date method 357

get\_full\_president\_data(), President method 369  
 get\_index\_of\_smallest\_number\_in\_array() method 267  
 get\_items\_of\_maker(), Collection method 487  
 get\_items\_of\_type(), Collection method 487  
 get\_planet\_name(), Planet method 321  
 get\_president\_name(), President method 369  
 get\_value(), IntParameter method 380  
 get\_week\_number(), Date method 358, 362  
 get\_year(), Olympics method 313  
 getBytes() String method 222  
 getChars() String method 222  
 getClass() method of class Object 435  
 getter methods 339  
 GIF, Graphics Interchange Format 48  
 golf, the game of 3  
 goto keyword 568  
 grams, unit of weight 319  
 graphical descriptions
 

- flowcharts 153
- graphical class descriptions 365

 greater than operator > 135  
 greater than or equal operator >= 135  
 Greenwich Mean Time (GMT) 527  
 Gregorian Calendar 347, 353  
 GregorianCalendar class 390, 521, 525, 526  
 GUI, graphical user interface 169

## H

hard disk 5  
 hardware 4  
 hashCode() method of class Object 435  
 header of a method 259  
 heap memory 326
 

- automatic management of 325

 heap memory vs. stack memory 189, 324  
 hectares, unit of area 319  
 hexadecimal input from the keyboard 120  
 hexadecimal literals 117, 163, 555  
 hexadecimal numbering system 33, 101  
 hexadecimal numbers, conventions for writing 39  
 hexadecimal output 206  
 hexadecimal-to-binary conversion 38, 534  
 hexadecimal-to-decimal conversion 36  
 hierarchies of classes 401, 404, 405, 415  
 high-level programming languages 94  
 HighClass, an example class 402  
 hyphenation 109

## I

i.e. means "in other words"  
 IBM Personal Computer 65  
 IC8 simulator 80  
 ICOM simulator 80  
 identifiers, see also "names" 108  
 if constructs (statements) 134–143, 560  
 if keyword 568  
 if-else if-else constructs 140

imaginary computer 54
 

- imaginary processor 56
- instruction decoding table 60
- internal registers 56
- machine instructions 59
- state diagram of operation 67
- vs. real computers 79

 IML language 73  
 IML programming
 

- address names 76
- keywords STRING, DATA, CONSTANT 78
- loop in a program 82
- loop inside another loop 88
- program that never terminates 96
- reserving memory locations 78
- stack and subroutine calls 86
- subroutine is called 84
- subroutine to read the keyboard 89

 immediate access memory 5  
 immediate subclass 400  
 immediate superclass 400  
 immutability of strings 204  
 implements keyword 514, 568  
 implicit calling of constructors 385  
 import keyword 568  
 import statements 334  
 in, static field of class System 333  
 inches, unit of length 319  
 increment operator ++ 155, 157  
 increment(), Date method 355  
 incremental program development 576  
 incremental software development 576  
 indentation of statements 167, 571  
 index variables 179, 407  
 index variables, naming of 188  
 index\_for\_day\_of\_week(), Date method 354  
 indexing 179  
 indexing, of strings 207  
 indexOf() String method 213, 223, 320  
 inheritance 384
 

- abstract classes and methods 424
- access modifiers 398
- automatic calling of superclass constructor 400
- constructors and inheritance 385, 386
- final classes 429
- interfaces inherited (extended) 520
- overriding methods 414
- polymorphic methods 414, 510
- subclasses and immediate subclasses 400
- superclasses and immediate superclasses 400

 initialization
 

- default values 192
- initialization of arrays 191
- initialization of data fields of a class 316
- initialization of multidimensional arrays 195
- initialization of variables 155
- initialized array of strings 231

 initialize\_account(), BankAccount method 296  
 initialize\_collection\_with\_test\_data() method 493

initialize\_rectangle(), Rectangle method 292  
 input parameters 261  
 InputMismatchException type 438  
 InputStream class 333  
 insert(), StringBuilder method 228  
 instance methods 291, 352  
 instance methods vs. class methods 208  
 instance variables 397  
 instanceof keyword (operator) 445, 568  
 instances (objects) of classes 291  
 int keyword 568  
 int type 14, 15, 100  
 integer (integral) variables 100, 105  
 integer literals 113  
 Integer, standard wrapper class 432, 433, 437  
 Integer.parseInt() method 351, 565  
 Integer.toHexString() method 565  
 integrated circuits 40  
 interface keyword 568  
 interfaces 514, 520  
 internal statements 152, 157, 259  
 international standard ISO 8601 354, 358  
 Internet Explorer browser 24  
 interpreter 28, 268  
 interrupt(), method of class Thread 546, 548  
 IntParameter class 380  
 IOException 482  
 is\_earlier\_than(), Date method 359  
 is\_equal\_to(), CollectionItem method 483  
 is\_equal\_to(), Date method 359  
 is\_last\_day\_of\_month(), Date method 352  
 is\_later\_than(), Date method 359  
 is\_not\_equal\_to(), Date method 359  
 is\_within\_dates(), Date method 353  
 iterations, see "loops"  
 iterators 508

**J**

jagged arrays 196  
 Jagger, Mick 519  
 jar tool 24  
 .java files, Java source program files 10  
 .java files, naming of 571  
 Java compiler  
     downloading from Internet 20  
 Java Development Kit (JDK) 20  
 Java documentation, electronic 22  
 Java interpreter 21, 28, 268  
 Java programs vs. traditional programs 10  
 Java runtime environment (JRE) 21, 28, 427  
 Java virtual machine 9, 21, 28, 268–272, 427  
 java.io package 460, 564  
 java.lang package 335  
 java.util package 335  
 javac, command to invoke Java compiler 25  
 JCreator 27–30  
 joules, unit of energy 319  
 JPEG, the format of Joint Photographic Experts Group 48

Julian Calendar 347

## K

keywords 108  
 keywords (reserved words), summary of 567  
 kilobits 39  
 kilograms, unit of weight 319  
 kilometers, unit of distance 111, 319

## L

L, suffix for long literals 117  
 laptop computers 65  
 lastIndexOf() String method 215, 223  
 leap years 342, 353  
 learning, discussion of v, 3  
 least significant bit 38, 549  
 least significant byte 104  
 least significant digit 32  
 left shift operator << 531  
 left-side expressions 558  
 left-to-right associativity 121  
 length of a string 204  
 length of an array 183  
 length() String method 204, 207, 223  
 length, data field 183  
 less than operator < 135  
 less than or equal operator <= 135  
 lexical analysis 131  
 lexical comparison of strings 220  
 light, speed of 115  
 lightyears, unit of distance 115, 319  
 Line Feed (LF) character 453  
 Linux computers and Java v  
 Linux operating system 4, 453, 538  
 literals  
     binary literals do not exist 530  
     boolean literals (true, false) 174  
     character literals 141, 555  
     floating-point literals 117, 555  
     hexadecimal literals 117, 163, 555  
     integer literals 113  
     literals of type float 192  
     literals of type long 117, 555  
     string literals 121, 203, 555  
 liters, unit of volume 319  
 little-endian order of bytes 106  
 load\_from\_file(), Collection method 489  
 local variables 275  
 locales 527  
 locating errors in programs 573  
 logical AND operator && 146  
 logical errors in programs 574  
 logical operators 145, 552  
 logical OR operator || 146  
 long keyword 568  
 long type 105  
 long type, literals 117, 555

Long, standard wrapper class 433

loops

body of a loop 159

do-while loops 164–165, 561

endless loop 154

for loops 158–163, 276, 561

"foreach" loops 176, 508

loops that are never entered 154, 574

programming style 571

while loops 152–157, 561

LowClass, an example class 403

lowercase letters 22, 45, 532

## M

machine code 9

machine instructions 59, 268

Macintosh computers 4

Macintosh computers and Java v

Madonna 519

main memory 5, 44, 50, 54

main storage 5

main() method 14, 495

invocation of method main() 272

parameters for method main() 278

placing of the main() method 293

mainframe computers 65

make\_speak(), Animal method 308

mantissa 110

Marley, Bob 519

masks and bit operators 532

matches() String method 223

Math class provides mathematical methods 446

mean value, calculation of 164

megabits 39

megabytes 39, 44

member selection operator, the dot operator . 559

MemberClass, an example class 402

members of classes

(data) fields 290

access modifiers 298, 398

accessor methods 313

constants 370

constructors 300

instance methods 291

order of members 316

memory

allocation of memory 188, 190, 272

array of integers in memory 182

deallocation of memory 272

heap memory 189

int variable in memory 104

objects in memory 295, 299

stack memory 189

String object in memory 202, 209

memory addresses, numerical 51

memory areas in main memory 189

memory cell 40, 44

memory locations 51, 52

memory management 301

memory management, automatic 325

menu-based program 494

meters, unit of distance 115, 319

methods

abstract methods 422

accessor methods 313, 339, 343, 351, 365

actual parameters 250, 268

arrays as method parameters 257

body of method 259

boolean methods 490

calling a method 240, 513

calling another instance method 352

class methods vs. instance methods 208

conventions for writing method names 252

final methods 429

formal parameters 250, 268

general structure of Java methods 259

getter methods 339

input parameters 261

instance methods 291

method declarators 244, 259

method header 259

methods taking parameters 244

mutator methods 339

native methods 272

object references as parameters 379

output parameters 260

outputting data to caller 254

overloaded method names 282

parameters passed by reference 245

parameters passed by value 245

polymorphic methods 414

public methods 262

setter methods 339

signature of a method 259

some basic method structures 562

stack usage in method calls 268–273

static methods 208, 397

typed methods 254, 255

typeless void methods 254, 255

Microsoft Windows 4

Microsoft Word 4

MiddleClass, an example class 402

miles, unit of distance 111, 319

minicomputers 65

modulus operator %, see "remainder operator" 138

month(), Date method 351

months, history of 237

moon, distance from Earth 117

moon, first moon landing 344

Morse codes 508

most significant bit 536

most significant digit 32

mouse, an input device 14

move(), Window method 408

MPEG, Moving Pictures Experts Group 48

MS-DOS operating system 4, 19, 65

MS-DOS window 19

- multi-line statements 109
- multidimensional arrays 196, 406
- multidimensional arrays, initialization of 195
- multiplication operator \* 113
- multiplicative arithmetic operators 559
- multiply and assign operator \*= 559
- multiprint\_character() method 248
- multitasking operating systems 538
- mutable strings 226
- mutator methods 339

## N

- names, rules for writing a name 109
- naming
  - .java files 571
  - constants 498
  - index variables 188
  - method names 252
  - names of methods that are of type bool 352
  - references to objects 238
- native keyword 568
- native methods 272
- natural names 108, 109
- natural naming 572
- nested program blocks 276
- new operator (keyword) 188, 291, 568
- newline character \n 102, 203, 225, 555
- nextByte(), Scanner method 118
- nextDouble(), Scanner method 118
- nextFloat(), Scanner method 118
- nextInt(), Scanner method 102, 120
- nextLine(), Scanner method 120, 205, 490
- nextLong(), Scanner method 118
- nextShort(), Scanner method 118
- non-negative variables 105
- not equal operator != 135
- NOT operator ! 146
- not\_in\_use, parameter for method main() 12, 278
- Notepad editor 27
- notes displayed by compiler 573
- NULL character \0 555
- null keyword 231, 568
- null references 231, 323
- numbering systems 32, 101
  - binary numbers 33
  - binary-to-decimal conversion 36
  - comparison 34
  - decimal-to-binary conversion 37
  - hexadecimal numbers 33, 101
  - hexadecimal-to-binary conversion 38
  - hexadecimal-to-decimal conversion 36
  - octal numbers 33
- numbers, important in computing 35
- numerical memory addresses 51

## O

- Object class 434, 436, 503

- object-oriented design
  - data encapsulation 339
- objects
  - creation of objects 300
  - destruction of objects in heap memory 301
  - initialization of objects 300
  - object creations 557
  - object references 291
  - objects of type String 200
  - referencing objects 295, 300, 404, 427
  - vs. object references 238
- octal numbering system 33
- odd numbers vs. even numbers 197
- Olympics, an example class 313
- opening files 451, 456, 472
- operands 117
- operating system, calling method Main() 272
- operating system, the role of 6
- operating systems 4, 538
- operators
  - addition operator + 113, 121
  - all used operators summarized 559
  - arithmetic operators 113, 138, 280
  - assignment operator = 113, 115
  - associativity of operators 121
  - bit operators 530–537
  - bitwise AND operator & 531
  - bitwise logical operators 552
  - bitwise OR operator | 531
  - comparison operators 135
  - complement operator ~ 531
  - concatenation operator + 121
  - conditional logical operators 552
  - decrement operator -- 157
  - division operator / 113
  - equal operator == 135
  - exclusive OR operator ^ 531
  - greater than operator > 135
  - greater than or equal operator >= 135
  - increment operator ++ 155, 157
  - instanceof operator 445
  - left shift operator << 531
  - less than operator < 135
  - less than or equal operator <= 135
  - logical AND operator && 146
  - logical operators 145, 552
  - logical OR operator || 146
  - modulus operator %, see "remainder operator" 138
  - multiplication operator \* 113
  - not equal operator != 135
  - NOT operator ! 146
  - operator && vs. operator & 552
  - operator || vs. operator | 552
  - precedence and associativity of operators 112
  - precedence of operators 145, 559
  - relational operators 135
  - remainder operator % 138, 280
  - right shift operator >> 531
  - subtraction operator - 113
  - unary operators 145

- zero fill right shift operator >>> 531
  - ounces, unit of weight 319
  - out, static field of class System 333
  - output parameters 260
  - output statements 12
  - output, formatting 116
  - outputting data to caller 254
  - overloading
    - method overloading vs. polymorphic methods 415
    - overloading constructors 306
    - overloading method names 282
  - overriding methods 414
- P**
- package keyword 568
  - packages 334, 398
    - default package 336
    - java.io package 460
    - java.lang package 335
    - java.util package 335
  - palindromes 206
  - palmtop computer 65
  - paper copies of programs 577
  - parameters 244–261
    - actual parameters 250, 268
    - arrays as method parameters 257
    - formal parameters 250, 268
    - input parameters 261
    - object references as parameters 379
    - output parameters 260
    - parameters for method main() 278
    - passing by reference 245
    - passing by value 245
    - to instance methods 296
    - varying in number 274
  - parameters vs. arguments 250
  - parentheses, meaning in expressions 113, 123
  - parseDouble(), method of class Double 433, 565
  - parseInt(), method of class Integer 351, 433, 565
  - parsing 433
  - percent sign % in format specifiers 125
  - Person, an example class 298
  - personal computers 65
  - pints, unit of volume 319
  - Planet, an example class 321
  - polymorphic methods 414, 504, 510
    - calling superclass version of a method 419
    - polymorphic methods vs. method overloading 415
  - pop(), Stack method 330
  - Pope Gregory XIII 347
  - pounds, unit of weight 319
  - precedence of operators 112, 145, 559
  - precision, of floating-point variables 107
  - prefix 0x in hexadecimal literals 117
  - President, an example class 368
  - PresidentInfoApplication, an example class 370
  - print() method 103
  - print(), abstract CurrentTime method 424
  - print(), AmericanTime method 425
  - print(), EnglishCalendar method 395
  - print(), European time method 425
  - print(), PrintStream method 333
  - print(), Window method 407
  - print\_all\_items(), Collection method 485
  - print\_array() method 506
  - print\_array(), overloaded method 284
  - print\_array\_of\_numbers() method 265
  - print\_calculations() method 280
  - print\_data\_of\_collection\_items() method 492
  - print\_data\_of\_next\_president() method 375
  - print\_in\_binary\_form() method 536
  - print\_lowercase\_letters() method 243
  - print\_message() method 241
  - print\_olympics\_data(), Olympics method 313
  - print\_person\_data(), Person method 298
  - print\_planet\_data(), Planet method 321
  - print\_rectangle(), Rectangle method 292
  - print\_sum() method 246
  - print\_text\_in\_decorated\_box() method 248
  - print\_uppercase\_letters() method 243
  - printf() method 116, 523
  - printf(), PrintStream method 333
  - printing date and time information 526
  - printing programs on paper 577
  - println(), PrintStream method 333
  - println(), PrintWriter method 455, 461
  - printStackTrace(), method of class Throwable 441
  - PrintStream class 333
  - PrintWriter class 454, 460
    - println() method 455, 461
  - private keyword 298, 398, 568
  - processes 538
  - processors 56
  - program blocks, nesting 276
  - program control 242
  - program counter 268, 269
  - program descriptions, structure of viii
  - program development, incremental 576
  - program editors 9, 27
  - program listings 577
  - program not working, what to do 574, 575
  - program pointer register 268, 269
  - program versions 577
  - programmers 9
  - programming exercises, list of xvii
  - programming exercises, practical advice for 29, 571
  - programming languages 8
    - assembly languages 94
    - high-level languages 94
  - programming style
    - see also "natural naming"
    - indentation of statements 167
    - loops and if constructs 571
    - order of class members 316
    - rules for using braces { } 167
  - programs
    - executable programs 8

- list of programs in this book xiv
- making backup copies 577
- printing on paper 577
- source code 9
- source programs 8
- protected keyword 298, 398, 416, 568
- pseudo-code 332
- public keyword 262, 298, 398, 569
- pure abstract classes 520
- push(), Stack method 329

## Q

quotes, see "double quotes" or "single quotes"

## R

- ragged arrays 196
- RAM, random access memory 5, 50
- random numbers, generation of 447
- read(), FileInputStream method 469, 470
- readability of programs 109
- readInt(), DataInputStream method 482
- readLine(), BufferedReader method 454, 459
- readUTF(), DataInputStream method 482
- recompiling programs 26
- Rectangle, an example class 292
- rectangular arrays 196
- redefined (rewritten) methods 414
- reference types 324
- reference types vs. value types 200
- references
  - object references 291
  - passing parameters by reference 245
  - string references 200
  - used as method parameters 379
  - vs. objects 238
- referencing objects 295, 300, 404, 427
  - this keyword 362
- referring to array elements 178
- regionMatches() String method 213, 224
- registers, in imaginary processor 56
- registers, in typical processor 269
- regular expressions 223, 234, 235
- relational expressions 141
- relational operators 135
- remainder operator % 138, 265, 280
- remove\_all\_items(), Collection method 485
- remove\_collection\_item() method 491
- remove\_one\_item(), Collection method 486
- repetitions, see "loops"
- replace() String method 215, 224, 465
- replace(), StringBuilder method 229
- replace\_string\_in\_file() method 465
- reserved words 108
- reserved words (keywords), summary of 567
- RestrictedAccount, an example class 419
- return keyword 569
- return statements 255

- reuse of methods 258
- right shift operator >> 531
- Roman numbers 33
- root directory 5
- run() method in a thread class 539, 541, 542, 544
- run(), CollectionMaintainer method 493
- run(), PresidentInfoApplication method 376
- running (executing) programs 4
- runtime environment 9, 21, 28, 427
- RuntimeException class 443

## S

- Scanner class 118, 120, 333, 522
  - findInLine() method 169
  - nextByte() method 118
  - nextDouble() method 118
  - nextFloat() method 118
  - nextInt() method 120
  - nextLine() method 120, 490
  - nextLong() method 118
  - nextShort() method 118
- Schwarzenegger, Arnold 519
- scope of variables 276, 561
- scope, class scope 277
- screen output, directing to file 399
- screen output, formatting 116
- search\_item\_with\_name(), Collection method 484
- search\_largest\_integer() method 256, 380
- search\_president\_by\_name() method 373
- search\_president\_for\_given\_date() method 374
- search\_string\_in\_file() method 462
- searching for errors in programs 574, 575
- selections, see "if constructs", "switch-case constructs"
- semicolon ; 14, 18
- serialization of objects 569
- server computer 65
- set(), ArrayList method 507
- set\_value(), IntParameter method 380
- setCharAt(), StringBuilder method 227
- setOut(), PrintStream method 333
- setter methods 339
- shallow copy of an object 434
- shift operations, shifting bits 534
- short keyword 569
- short type 105
- Short, standard wrapper class 433
- show\_account\_data(), BankAccount method 296, 302, 416
- signature of a method 259
- simulation 80
- simultaneous execution of programs 538
- sine of an angle 447
- single quote character ' 141, 203, 555
- size() method of ArrayList 503
- sleep(), method of class Thread 540, 546
- software 4
- software development, incremental 576
- sort(), Collections method 518
- sort\_to\_ascending\_order() method 266

- sorting algorithms 262
- source code 9
- source programs 8
- source programs, downloading 23
- SpanishCalendar, an example class 396
- specifications of exceptions 442
- speed of light 115
- split() String method 224, 234, 235
- stack
  - role of stack in method calls 268–273
  - stack implemented with software 326
  - stack pointer register 268
- stack frame 270
- stack memory vs. heap memory 189, 324
- Stack, an example class 328
- stack\_is\_not\_empty(), Stack method 330
- standard input device 118
- standard Java classes 434
- start(), method of class Thread 539, 541
- startsWith() String method 224
- statements
  - assignment statements 113, 115, 558
  - break statements 147, 150
  - do-while loops 164–165, 561
  - for loops 158–163, 561
  - if constructs 134, 560
  - import statements 334
  - internal statements 152, 157, 259
  - method calls 242
  - output statements 12
  - return statements 255
  - switch-case constructs 147–150, 560
  - the role of semicolon ; 100
  - throw statements 438
  - try-catch constructs 171, 173, 438
  - while loops 152–157, 561
- static binding 427
- static keyword 397, 569
- static methods 208, 397
- status variables (state variables) 498
- stop\_this\_thread() method 541, 542
- store\_text\_lines\_to\_file() method 464
- store\_to\_file(), Collection method 488
- store\_to\_file(), CollectionItem method 483
- string arrays 234
- String class 200
  - constructors 221
- string constants, see "string literals"
- string literals 121, 203, 555
- String methods 208, 220
  - charAt() method 204, 205, 220
  - compareTo() method 212, 220
  - compareToIgnoreCase() method 220
  - concat() method 221
  - contains() method 213, 221, 320
  - contentEquals() method 221
  - copyValueOf() method 221
  - endsWith() method 221
  - equals() method 210, 212, 222
  - equalsIgnoreCase() method 222
  - format() method 126, 222, 274, 360, 369
  - getBytes() method 222
  - getChars() method 222
  - indexOf() method 213, 223, 320
  - lastIndexOf() method 215, 223
  - length() method 204, 207, 223
  - matches() method 223
  - regionMatches() method 213, 224
  - replace() method 215, 224, 465
  - split() method 224, 234, 235
  - startsWith() method 224
  - substring() method 213, 216, 224
  - toCharArray() method 225
  - toLowerCase() method 218, 225, 323
  - toString() method 225
  - toUpperCase() method 225
  - trim() method 219, 225
  - valueOf() method 225
- StringBuffer class 230
- StringBuilder class 226, 474
  - append() method 229
  - delete() method 229
  - insert() method 228
  - replace() method 229
  - setCharAt() method 227
- StringIndexOutOfBoundsException 490
- strings
  - arrays of strings 231
  - concatenating String objects 214
  - creating String objects 216
  - dynamic (mutable) strings 226
  - empty string 204, 467
  - equality of String objects 210, 467
  - finding substrings in String objects 223
  - immutability of strings 204
  - length of a string 204
  - searching for substrings inside strings 213
  - String object in memory 202, 209
  - taking substrings from String objects 224
- strong typing 267
- subclasses 400
- subinterfaces 520
- subroutines 240
- subscripting, see "indexing"
- substring() String method 213, 216, 224
- substrings in String objects 223
- subtract and assign operator -= 559
- subtraction operator - 113
- suffix F identifies float literals 117
- suffix L identifies long literals 117
- Sun Microsystems 20
- Sun, radius of 498
- super keyword 385, 386, 419, 569
- superclass 384, 396
- supercomputers 65
- superstitious friends, how to help them 342
- switch keyword 569
- switch-case constructs 147, 150, 560

symbols, see also "names" 108  
 synchronized keyword 569  
 syntax of a programming language 12  
 System class 333  
 System.currentTimeMillis() method 544  
 System.in 333  
 System.out.print() method 103  
 System.out.printf() method 116, 125, 128, 523  
 systems 9

## T

tab, see "tabulator character"  
 table of contents x  
 tabulator character `\t` 203, 225, 555  
 tabulator character, avoiding 577  
 Task Manager in Windows 576  
 text balloons viii  
 text files 452  
 text files vs. binary files 469  
 TextWindow, an example class 410  
 this keyword 362, 363, 569  
 this\_is\_a\_leap\_year(), Date method 353  
 Thread class 541  
 Thread.sleep() method 540, 546  
 threads 538–548  
 ThreadToDisplayTicksAndTime, an example class 544  
 ThreadToPrintDollarSigns, an example class 542  
 ThreadToPrintDots, an example class 541  
 ThreadToShowFullTimeInfo, an example class 547  
 ThreadToShowSeconds, an example class 546  
 throw keyword 438, 569  
 throw statements 438  
 Throwable class 443  
 throwing exceptions 438  
 throws keyword 442, 482, 569  
 TIFF, Tagged Image File Format 48  
 time measurement 426  
 time printing formats 423  
 to\_american\_format\_string() method 386  
 to\_line\_of\_text(), CollectionItem method 483  
 to\_string\_with\_month\_name() method 386  
 toCharArray() String method 225  
 toHexString(), method of class Integer 565  
 toLowerCase() String method 218, 225, 323  
 toString() method in class Event 516  
 toString() method of class Object 435, 504  
 toString() String method 225  
 toString(), Date method 360  
 toUpperCase() String method 225  
 traditional programs vs. Java programs 10  
 transfer\_money\_to(), BankAccount method 303, 417  
 transient keyword 569  
 transistors 40  
 translate(), a polymorphic method 510, 511, 512  
 TrilingualTranslation, an example class 511  
 trim() String method 219, 225  
 true, boolean literal 174, 569  
 truth values 134, 174

try keyword 171, 569  
 try-catch constructs 171, 173, 438  
 try-catch-finally constructs 444, 457  
 Turing machine 65  
 Turing, Alan 207  
 .txt, name extension for text files 452  
 type parameter of generic type 502  
 type parameters of ArrayList arrays 505  
 typed methods 254, 255  
 typeless void methods 254, 255  
 types  
   explicit type conversions 161, 164, 267  
   reference types vs. value types 200, 324  
 typing with 10 fingers vi

## U

UML, Unified Modeling Language 365  
   describing inheritance between classes 389  
   examples of class diagrams 389, 401, 405, 415  
   has-a relationship between classes 367  
 unary operators 145  
 unboxing 432  
 unchecked exceptions 442, 443  
 underscore character `_` 108  
 unhandled exceptions 120  
 Unicode character coding 46, 47, 105  
 Unicode Transformation Format 456  
 Universal Time 527  
 UNIX computers and Java v  
 UNIX operating system 4, 453, 538  
 unsigned variables 105  
 uppercase letters 22, 45, 532  
 UTF-8 456, 477

## V

value types 324  
 value types vs. reference types 200  
 valueOf() methods 565  
 valueOf() String method 225  
 varargs 274  
 variables  
   boolean type 105  
   byte type 105  
   char type 105  
   classwide variables 275  
   double type 110  
   float type 110  
   floating-point variables 110  
   initialization of variables 155  
   instance variables vs. class variables 397  
   int type 100  
   local variables 275  
   long type 105  
   non-negative variables 105  
   parameter variables 247  
   short type 105  
   table of all types 107, 433

- variable declarations 100, 556
- varying number of parameters 274
- versions of programs 577
- virtual keyword (C++ and C#) 429
- virtual machine 9, 28, 268–272, 427
- visibility modifiers 398
- visibility of class members 298
- void keyword 254, 569
- volatile keyword 569
- vs. means "versus" or "against"

## W

- warnings given by compiler 573
- was\_president\_on(), President method 369
- weak typing 267
- week numbers 358
- while keyword 569
- while loops 152–157, 561
- whitespace characters 225
- whole numbers (integers) 14
- Window, an example class 406
- Windows operating system 4
- Windows XP operating system 538
- windows, character-based 404
- Windows/MS-DOS operating system 453
- withdraw\_money(), BankAccount method 303, 417
- withdraw\_money(), redefined method 418, 419
- workstation 65
- World Calendar 347
- write(), FileOutputStream method 471
- writeInt() and other DataOutputStream methods 476
- writeInt(), DataOutputStream method 482
- writeUTF(), DataOutputStream method 477, 483
- writing about objects and their references 238

## Y

- yards, unit of distance 115, 319
- year(), Date method 351
- year, definition of 342

## Z

- zero fill right shift operator >>> 531
- .zip files, unzipping (extracting files) 23

## USEFUL TABLES

On the following pages, you will find tables that are useful when you work with computer programs. Don't hesitate to use this page to write down important information that you need to remember in your programming work.

## Character codes from 0 to 127.

DEC	HEX	CHARACTER	DEC	HEX	CHAR	DEC	HEX	CHAR
0	00	NULL	48	30	0	96	60	`
1	01		49	31	1	97	61	a
2	02		50	32	2	98	62	b
3	03		51	33	3	99	63	c
4	04		52	34	4	100	64	d
5	05		53	35	5	101	65	e
6	06		54	36	6	102	66	f
7	07	BELL	55	37	7	103	67	g
8	08	BACKSPACE	56	38	8	104	68	h
9	09	TABULATOR	57	39	9	105	69	i
10	0A	NEWLINE, LF	58	3A	:	106	6A	j
11	0B		59	3B	;	107	6B	k
12	0C		60	3C	<	108	6C	l
13	0D	CR	61	3D	=	109	6D	m
14	0E		62	3E	>	110	6E	n
15	0F		63	3F	?	111	6F	o
16	10		64	40	@	112	70	p
17	11		65	41	A	113	71	q
18	12		66	42	B	114	72	r
19	13		67	43	C	115	73	s
20	14		68	44	D	116	74	t
21	15		69	45	E	117	75	u
22	16		70	46	F	118	76	v
23	17		71	47	G	119	77	w
24	18		72	48	H	120	78	x
25	19		73	49	I	121	79	y
26	1A		74	4A	J	122	7A	z
27	1B	ESCAPE	75	4B	K	123	7B	{
28	1C		76	4C	L	124	7C	
29	1D		77	4D	M	125	7D	}
30	1E		78	4E	N	126	7E	~
31	1F		79	4F	O	127	7F	DELETE
32	20	SPACE	80	50	P			
33	21	!	81	51	Q			
34	22	"	82	52	R			
35	23	#	83	53	S			
36	24	\$	84	54	T			
37	25	%	85	55	U			
38	26	&	86	56	V			
39	27	'	87	57	W			
40	28	(	88	58	X			
41	29	)	89	59	Y			
42	2A	*	90	5A	Z			
43	2B	+	91	5B	[			
44	2C	,	92	5C	\			
45	2D	-	93	5D	]			
46	2E	.	94	5E	^			
47	2F	/	95	5F	_			

In the first column:  
LF = Line Feed  
CR = Carriage Return

You get the 2-byte (16-bit) Unicode character codes of these 1-byte codes if you add 8 zero bits before each code.

**Numbers expressed in different numbering systems.**

DEC	HEX	BINARY	DEC	HEX	BINARY
0	00	0000 0000	48	30	0011 0000
1	01	0000 0001	49	31	0011 0001
2	02	0000 0010	50	32	0011 0010
3	03	0000 0011	51	33	0011 0011
4	04	0000 0100			
5	05	0000 0101	...		
6	06	0000 0110			
7	07	0000 0111	61	3D	0011 1101
8	08	0000 1000	62	3E	0011 1110
9	09	0000 1001	63	3F	0011 1111
10	0A	0000 1010	64	40	0100 0000
11	0B	0000 1011	65	41	0100 0001
12	0C	0000 1100	66	42	0100 0010
13	0D	0000 1101	67	43	0100 0011
14	0E	0000 1110			
15	0F	0000 1111	...		
16	10	0001 0000			
17	11	0001 0001	126	7E	0111 1110
18	12	0001 0010	127	7F	0111 1111
19	13	0001 0011	128	80	1000 0000
20	14	0001 0100	129	81	1000 0001
21	15	0001 0101	130	82	1000 0010
22	16	0001 0110			
23	17	0001 0111	...		
24	18	0001 1000			
25	19	0001 1001	254	FE	1111 1110
26	1A	0001 1010	255	FF	1111 1111
27	1B	0001 1011	256	100	0001 0000 0000
28	1C	0001 1100	257	101	0001 0000 0001
29	1D	0001 1101	258	102	0001 0000 0010
30	1E	0001 1110			
31	1F	0001 1111	...		
32	20	0010 0000			
33	21	0010 0001	510	1FE	0001 1111 1110
34	22	0010 0010	511	1FF	0001 1111 1111
35	23	0010 0011	512	200	0010 0000 0000
36	24	0010 0100	513	201	0010 0000 0001
37	25	0010 0101			
38	26	0010 0110	...		
39	27	0010 0111			
40	28	0010 1000	1022	3FE	0011 1111 1110
41	29	0010 1001	1023	3FF	0011 1111 1111
42	2A	0010 1010	1024	400	0100 0000 0000
43	2B	0010 1011	1025	401	0100 0000 0001
44	2C	0010 1100	1026	402	0100 0000 0010
45	2D	0010 1101			
46	2E	0010 1110	...		
47	2F	0010 1111			

**Important numbers in computing.**

$2^n$	DECIMAL	HEX	"slang"
$2^0$	1	1H	
$2^1$	2	2H	
$2^2$	4	4H	
$2^3$	8	8H	
$2^4$	16	10H	
$2^5$	32	20H	
$2^6$	64	40H	
$2^7$	128	80H	
$2^8$	256	100H	
$2^9$	512	200H	
$2^{10}$	1024	400H	1 k
$2^{11}$	2048	800H	2 k
$2^{12}$	4096	1000H	4 k
$2^{13}$	8192	2000H	8 k
$2^{14}$	16384	4000H	16 k
$2^{15}$	32768	8000H	32 k
$2^{16}$	65536	10000H	64 k
$2^{17}$	131072	20000H	128 k
$2^{18}$	262144	40000H	256 k
$2^{19}$	524288	80000H	512 k
$2^{20}$	1048576	100000H	1 M
$2^{21}$	2097152	200000H	2 M
$2^{22}$	4194304	400000H	4 M
$2^{23}$	8388608	800000H	8 M
$2^{24}$	16777216	1000000H	16 M
$2^{25}$	33554432	2000000H	32 M
$2^{26}$	67108864	4000000H	64 M
$2^{27}$	134217728	8000000H	128 M
$2^{28}$	268435456	10000000H	256 M
$2^{29}$	536870912	20000000H	512 M
$2^{30}$	1073741824	40000000H	1 G

These are sample pages from Kari Laitinen's book  
 "A Natural Introduction to Computer Programming with Java".  
 For more information, please visit  
<http://www.naturalprogramming.com/javabook.html>