messages, 344
servlets, 304–305
final, 57–58, 60, 443
finalize(), 443
finally, 136, 137–138, 225, 444
findThreeInARow(), 117–118
firewalls, 189
first-in-first-out (FIFO), 146, 150, 229, 337
Float, 255
float, 20, 255
FlowLayout, 77–79
flush(), 166
FocusListener, 96–97
for loop, 42, 69, 158
forEach, 320
for-each, 44, 144, 146
@FormParam, 390
forward(), 317
FROM, 378
functions, 5
hash, 147
method signatures, 31
Future, 226–227
get(), 227, 368
javax.ejb.AsyncResult, 367
FX.deferAction(), 426

G

GameConstants, 124, 127–128
garbage collector (GC), 21, 36, 440
GC. *See* garbage collector
@GeneratedValue, 376
GenerationType.IDENTITY, 377
generics, 155–162
casting, 156, 158–159
get(), 147
methods, 160–161
myData, 255
super, 160
geonames.org, 392
Get, 301, 441
get(), 145
Future, 227, 368
generics, 147
multi-threading, 228
@GET, 391
getBean(), 400
getColumnClass(), 257
getColumnCount(), 252
getColumnName(), 256
getConnection(), 356
getConstructors(), 269
getContent(), 205
getDisplayValue(), 95
getFields(), 269
getFrom(), 205
getinstance(), 370
getInterfaces(), 269
getJSON(), 396
getMessage(), 136
getParameter(), 301
getPreferredSize(), 127
getProperty(), 148
getRegistry(), 277
getResultList(), 380
getRowCount(), 253
getSession(), 303–304
getSingleResult(), 380
getSource(), 92
getState(), 219
getStore(), 204
getSubject(), 205
getTableCellRendererComponent(), 258
getter methods, 95, 242
getText(), 94
getValueAt(), 252, 253
GlassFish, 287, 344
Eclipse, 297
EJB, 366–367
JavaFX, 424
JNDI, 350–355, 424
naming service, 352–355
global variables, 148
Goetz, Brian, 224
Gosling, James, 1
graphic user interfaces (GUIs), 75–88, 115, 230–231
Graphics, 127
GridBagLayout, 84–85

Page 493 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

GridLayout, 79–81
GROUP BY, 378
GroupLayout, 115
GUIs. *See* graphic user interfaces

H

hash code, 147–148
hash function, 147
HashMap, 41, 147–149, 445
HashSet, 146
Hashtable, 147–149, 352, 445
hasMoreElements(), 149
<head>, 105, 392
height, 106
Heiss, Janice, 209
Hello World, 4–7
    compile, 6–7
    Eclipse, 11–15
Hibernate, 235, 411–419
    DBMS, 413–515
    Eclipse, 412–413
    installation, 412–415
    Java EE, 289
    retrieving data, 415–417
    Spring Framework, 401
Hibernate Query Language (HQL), 412
host name, 185, 200
host_name, 279
Hotspot JVM, 58
HQL. *See* Hibernate Query Language
.htm, 105
HTML, 105–107, 401–408
    JavaFX, 429–432
    JSP, 310–311, 323
    wrappers, 114–115
.html, 105
HTTP. *See* Hypertext Transfer Protocol
http.proxyHost, 189
http.proxyPort, 189
HTTPRequest, 293
HttpRequestListener, 305–306
HTTPServlet, 295, 301
HTTPServletRequest, 301
    cookies, 302–303
    getParameter(), 301
    request, 311
HTTPServletResponse, 294, 301, 311, 317
HTTPSession, 311, 444
HttpSessionListener, 305–306
HttpURLConnection, 189
Hypertext Transfer Protocol (HTTP), 293, 301–302, 445, 446
    proxy servers, 189
    stateless protocol, 302

I

@Id, 376
IDEs. *See* Integrated Development Environments
if, 25, 49–50, 131
    else, 49
    null, 44, 45, 444
    polymorphism, 69
if-else, 49
ij, 236–237
IllegalCastException, 145, 155
IMAP. *See* Internet Message Access Protocol
import, 56
include, 312, 313
include(), 317
index.html, 392
inheritance
    applets, 71
    classes, 26–27, 60, 69, 440
    multi-threading, 212
@inherited, 264
init(), 107–108, 134
    filters, 304–305
    servlets, 301
initial context, 349
InitialContext, 353
inner classes, 98–100
input/output (I/O), 134–135, 163, 186
InputStream, 164–165
InputStreamReader, 188
inputText, 330

Page 494 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

`INSERT`, 237
`Insert`, 247
instances
  classes, 18
    ancestors, 60
    interfaces, 61
    `static`, 21
  JVM, 441
  methods, 33
    classes, 100
  objects, 36
  variables, 36–37
`instanceof`, 60
  `ArrayList`, 145–146
  casting, 92
  generics, 158
  polymorphism, 69
`int`, 20, 255
`Integer`, 255
Integrated Development Environments (IDEs)
  debugging, 46
  Eclipse, 9–15
    stack trace, 134
  NetBeans, 87
  plug-ins, 297
interceptors, 289
interfaces, 58–60. *See also specific interfaces*
  abstract classes, 70–71, 439
  ancestor, 70–71
  classes, 58–60
    abstract, 70–71
    instances, 61
  collections, 146
  declarations, 58
  method signatures, 59
  polymorphism, 69–70, 71
  RMI, 274–276
Internet Message Access Protocol (IMAP), 199
Internet service provider (ISP), 185
`interrupt()`, 214, 215
inversion of control (IoC), 399
`invoke()`, 269
`invokeLater()`, 231, 426, 441
I/O. *See* input/output
IoC. *See* inversion of control
`IOException`, 135, 136, 137, 216, 442
IP address, 185–186, 193
`isAlive()`, 219
`isAnnotationPresent()`, 269
ISP. *See* Internet service provider
`ItemListener`, 96–97
`Iterable`, 146
`Iterator`, 149–151

J

`.jar`, 12
Java API for RESTful Web Services (JAX-RS), 288, 388
Java architect, 438
Java Collections, 149
Java Community Process (JCP), 284
Java Concurrency in Practice (Goetz), 224
Java Control Panel, 104
Java Development Kit (JDK), 1–4, 96–97
Java EE, 283–291
  Eclipse, 10–11
  e-mail, 199–200
  Hibernate, 289
Java Enterprise Edition. *See* Java EE
Java Messaging Service (JMS), 284, 335–347, 350
  API, 337–338
  EJB, 362
  queues, 335
  Resources, 350
Java Naming and Directory Interface (JNDI), 349–359, 442
  `DataSource`, 355–357
  EJB, 364
  GlassFish, 350–355, 424
  LDAP, 357–358
Java Network Launch Protocol (JNLP), 104, 430
Java Persistence API (JPA), 287, 375–386
Java Persistence Query Language (JPQL), 375, 376, 378
Java Rich Internet Applications Deployment Advice, 106

Page 495 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

Java Runtime Environment (JRE), 3–4, 12
Java SE. *See* Standard Edition
Java Specification Request (JSR), 284–285
Java Transaction API (JTA), 287
Java Virtual Machine (JVM), 7
  applets, 103
  arrays, 51
  `CharSet`, 166
  errors, 135
  GC, 36
  instances, 441
  libraries, 12
  multi-threading, 214
  RMI, 273
  serialization, 182
`java.awt`, 97, 103, 107
`java.awt.EventQueue`, 230
JavaBeans, 315–317
`javac`, 6–7
JavaDB. *See* Derby
javadoc, 22
JavaFX, 75, 285, 421–433
  applets, 424, 430
  GlassFish, 424
  HTML, 429–432
  JNLP, 430
  `Messages`, 424
  Open MQ, 424
  reading code, 426–429
  SDK, 426
`JavaFXMessageHandler`, 426
`javafx-ui.jar`, 430
`java.io`, 163
`java.lang`, 56, 262
`java.lang.annotation`, 262
`java.lang.Enum`, 219
`java.lang.reflect`, 268
`java.lang.RMISecurityManager`, 278
`java.lang.SecurityManager`, 278
JavaMail, 199
`java.net`, 186
`java.net.Proxy`, 189
`java.policy`, 278
`java.rmi.registry.Registry`, 279
JavaScript Object Notation (JSON), 392
JavaServer Face (JSF), 285, 324–325
  converters, 333
  MVC, 445
  templates, 326–331
  UI, 323
  validators, 333
  Web applications, 323–334
JavaServer Pages (JSP), 284, 310–321
  deploying, 445
  HTML, 310–311, 323
  JavaBeans, 316
  navigation, 331–332
  servlets, 317–318, 444
  tags, 312–314, 318–320
  variables, 311–312, 444
`java.sql`, 235
`java.util`, 41, 143, 146
`java.util.concurrent`, 143, 224–230
`java.util.List`, 157
`java.util.logging`, 46
`javax.annotation`, 262
`javax.ejb.AsyncResult`, 367
`javax.faces`, 325
`javax.mail.internet.InternetAddress`, 200
`javax.mail.Message`, 200
`javax.mail.Session`, 200
`javax.mail.Transport`, 200
`javax.naming.InitialContext`, 352
`javax.servlet`, 296
`javax.servlet.Filter`, 304
`javax.servlet.http.HttpSession`, 303
`javax.sql`, 235
`javax.sql.DataSource`, 356
`javax.swing.SwingWorker`, 231
`javax.validation.constraints`, 333, 382
JAX-RS. *See* Java API for RESTful Web Services
JBoss, 287
`JButton`, 76, 92, 230, 251
`JComboBox`, 170
JCP. *See* Java Community Process
JDBC, 241–248, 443
  connection pools, 247–248
  databases, 235–249

Page 496 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

ODBC, 236
sample program, 238–240
transactional updates, 247
JDK. *See* Java Development Kit
Jetty, 287
`JFrame`, 76–77
Jigloo GUI Builder plug-in, 88
`JLabel`, 126, 258
`JList`, 76
JMS. *See* Java Messaging Service
`JMSListener`, 424
JNDI. *See* Java Naming and Directory Interface
JNLP. *See* Java Network Launch Protocol
`JOB_TITLE`, 237
Johnson, Rod, 399
`Join`, 381
`join()`, 223–224
`JOptionPane`, 91
JPA. *See* Java Persistence API
`JPanel`, 76
JPQL. *See* Java Persistence Query Language
JQuery, 392
JRE. *See* Java Runtime Environment
JSF. *See* JavaServer Face
JSON. *See* JavaScript Object Notation
JSP. *See* JavaServer Pages
`.jsp`, 404
JSP Standard Tag Libraries (JSTL), 313, 320
`<jsp:param>`, 314
`_jspService()`, 313
`JspWriter`, 311
JSR. *See* Java Specification Request
JSTL. *See* JSP Standard Tag Libraries
JTA. *See* Java Transaction API
`JTable`, 76
callback methods, 253–257
MVC, 251–257
Swing, 251–260
`JTextField`, 94, 95
JVM. *See* Java Virtual Machine

K

`KeyAdapter`, 97–98
`keyFrames`, 429
`KeyListener`, 96–97, 130
key/value pairs, 146–148, 157, 302, 338
King, Gavin, 411
`<K,V>`, 157

L

`Label`, 429
LAN. *See* local area network
`lastModified()`, 172
late binding, 68
layout managers, 77–88
lazy loading, 412
LDAP. *See* Lightweight Directory Access Protocol
`length()`, 172
libraries, 2, 12, 58
`lib/security`, 278
life cycle, 2
Lightweight Directory Access Protocol (LDAP), 357–358
`LIKE`, 381
`LinearGradient`, 429
`LinkedBlockingDeque`, 229
`LinkedBlockingQueue`, 229
`LinkedList`, 41
collections, 150–151
`List`, 146
links, 2
`List`, 41, 146
`list()`, 172
listeners. *See* event listeners
`ListIterator`, 150, 151
LMaven, 296
`load()`, 148
`loadFromXML()`, 149
`@Local`, 364, 365
local area network (LAN), 185, 279
local beans, 364–366
`@LocalBean`, 363
`LOCAL_VARIABLE`, 265
`@Lock`, 370
locks, 216–217, 225
`Logger`, 46
`long`, 20
loops, 42–46, 69, 158
loosely coupled objects, 221

Page 497 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

M

MAC OS X plug-ins, 104
Mailer, 201–204
main, 51
main(), 5–6, 24–25, 134, 201
  applets, 107
  multi-threading, 211
main-class, 430
Main.fx, 430
MalformedURLException, 186
managed beans, 324–325, 333
@ManagedBean, 325
Map, 146, 147
MapMessage, 338
marker interfaces, 60, 176
Matisse plug-in, 87
@Max, 333
MAX_PRIORITY, 216
MDBs. *See* message-driven beans
member variables, 21
Message, 338
messages
  filters, 344
  publishing, 343
  receiving, 340–342
  sending, 338–341
  subscribers, 343–344
message consumers, 337, 340
message producers, 337
@MessageDriven, 363
message-driven beans (MDBs), 286, 340, 362, 371, 446
MessageHandler, 424
MessageListener, 340
message-oriented middleware (MOM), 286, 335–347
  asynchronous communication, 335
  DBMS, 337
  EJB, 362
  queues, 336
MessageProducer, 338
Messages, 424
MessageSender, 353
messageService, 430
meta-annotations, 265
metadata, 181, 261–272, 296
META-INF, 296, 299, 363
METHOD, 265
methods, 5–6, 57
  arguments, 31
  classes, 17, 23–24, 100
    declarations, 31–39, 58
  event listeners, 98
  generics, 160–161
  getter, 95
  Hotspot JVM, 58
  instances, 33
    classes, 100
  main, 51
  objects, 69
  overloads, 31–32
  overrides, 27, 98
  setter, 95
  signatures, 5–6, 31, 59
  static, 38
  values, 36–37
Method.getParameterTypes(), 269
Microsoft Active Directory, 357
middleware, 236, 286, 335–347
MIME. *See* Multipurpose Internet Mail Extensions
@Min, 333
MIN_PRIORITY, 216
mkDir(), 172
ModelAndView, 405
model-view-controller (MVC), 170, 251–257, 441
  JSF, 324, 445
  Spring Framework, 399–409
MOM. *See* message-oriented middleware
mouseClicked(), 116, 129
MouseEvent, 129
MouseListener, 96–97, 115, 129, 131
MouseMotionListener, 96–97, 129, 131
mouseMoved(), 129
Multipurpose Internet Mail Extensions (MIME), 199, 392
multi-threading, 209–222
  call(), 228
  collections, 230

Page 498 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

concurrent execution, 223–233
EJB, 362
executor framework, 225–229
get(), 228
inheritance, 212
IOException, 216
join(), 223–224
Object, 219–222
priorities, 216
queues, 229–230
race conditions, 216–219
runnable interfaces, 211–212
subclasses, 225
Swing, 230–233
synchronization, 216–219, 441
wait(), 219–222
MVC. *See* model-view-controller
myData, 255
MyEclipse, 87
MyFrame, 253
MyMailAuthenticator, 201

N

name, 296
name_service_port, 279
namespaces, 327–328
naming service, 279, 349, 352–355
navigation, 331–332
NetBeans IDE, 87
networks, 185–198
HTTP proxy servers, 189
sockets, 193–197
stock quote program, 191–193
NEW, 219
newFixedThreadPool(), 226
newTextlabel, 429
next(), 150, 246
nextElement(), 149
non-durable subscribers, 343
NORM_PRIORITY, 216
NoSuchMethodException, 270
NOT operation, 49–50
notify(), 219–222, 225, 440
notifyAll(), 219–222, 225, 440
notifyOnUpdates, 266
@NotNull, 333, 377
NULL, 243
null, 43, 44, 45, 147, 269, 444
NullPointerException, 135, 444

O

Object, 92
get(), 145
inheritance, 60
multi-threading, 219–222
<object>, 106
objects
cloning, 183
communication, 93–96
DTOs, 241–242
exceptions, 139
instances, variables, 36
loosely coupled, 221
marker interfaces, 60
methods, 69
OO, 17–18
RMI, 279
object oriented (OO), 2, 17–29
ObjectInputStream, 178–179
ObjectMessage, 338
ObjectOutputStream, 176–178, 180
object-relational mapping (ORM), 235, 375, 411
ODBC, 236
oldTextlabel, 429
onClose, 429
one-to-many, 343
onMessage(), 340, 342, 371
OO. *See* object oriented
OpenDS, 357
OpenLDAP, 357
Open MQ, 336, 344–346, 424
operating system (OS), 3, 75
OR operation, 49
Oracle Directory Server, 357
ORDER BY, 378
ORM. *See* object-relational mapping
OS. *See* operating system
OUT, 243

Page 499 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

out, 311
out of memory error, 445
outer classes, 99
output directory, 12
OutputStream, 164–165, 301
outputText, 330
overloads, 31–32, 35
@Override, 231, 263
overrides, 27, 57, 98

P

P2P. *See* point-to-point
PACKAGE, 265
package, 14, 57
packages, 13–14, 55–56, 95, 441
page, 312
pageContext, 312
paint(), 107–108, 127
paintComponent(), 127
<param>, 106
PARAMETER, 265
passivation, 176, 363
PATH, 236
persist(), 379
persistence units, 379–380
ping-pong, 123–132
PingPongGameEngine, 124, 128–130
PingPongGreenTable, 130
plain old Java object (POJO), 286, 324, 361–363
PLAIN_MESSAGE, 91
platform independence, 1, 3
plugin, 314
plug-ins. *See also specific plug-ins*
    applets, 103–104
    Eclipse, 9
    IDEs, 297
    Jigloo GUI Builder, 88
    MAC OS X, 104
    Matisse, 87
point, 129
point-to-point (P2P), 337
POJO. *See* plain old Java object
polymorphism, 60, 68–71
POP. *See* Post Office Protocol
ports, 193, 200
Post, 301, 441
Post Office Protocol (POP), 199
@PostConstruct, 370
@PostPersist, 382
@PostRemove, 382
PreparedStatement, 242–243
presentation layer, 193, 300
presentation tier, 285
previous(), 150
primary keys, 376
primitive data types, 19–20
printIn(), 151
println(), 5–6
printStackTrace(), 136
PrintWriter, 300, 301
priority queues, 146
PriorityBlockingQueue, 229
private, 57, 95, 131
process(), 232
processData(), 158
profiles, 288
projects, 10
properties, 18, 22–23
Properties, 148–149, 352
protected, 57
proxy
    RMI, 273
    servers, 189
pruning, 288
public, 5, 57
public API, 94–96
public interface, 56
publish(), 232, 343
publish/subscribe mode, 337
pushes, 307
PUT, 388

Q

Quartz Scheduler, 372
@QueryParam, 390
Queue, 146, 337, 338, 340

Page 500 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

queues
  FIFO, 229
  JMS, 335
  MOM, 336
  multi-threading, 229–230
  priority, 146
  unbound, 229
QueueConnection, 337
QueueConnectionFactory, 337, 340
QueueReceiver, 337, 340
QueueReceiver.receive(), 340
QueueSender, 337, 339
QueueSession, 337

R

race conditions, 216–219
RCP. *See* Rich Client Platform
RDBMS. *See* Relational Database Management Systems
read(), 135, 137, 138
readExternal(), 180
readObject(), 178
READ_ONLY, 205
Read_WRITE, 205
receive(), 340
record fields, 235
Rectangle, 429
ReentrantLock, 225
reflection, 267–270, 445
registry, 273, 276–277
regular methods, 440
Relational Database Management Systems (RDBMS), 235
@Remote, 364
remote beans, 366
Remote Method Invocation (RMI), 273–281
  applets, 279
  clients, 273, 277–278
  JVM, 273
  proxy, 273
  registry, 276–277
  security, 278–279
remote procedure calls (RPC), 335
RemoteException, 141
remove(), 379
@Remove, 369
renameTo(), 172
renderers, 257–259
repaint(), 127, 130
representation state of transfer (REST), 388
request, 311, 313, 316, 317
RequestDispatcher, 317
request.getsSession(), 302
@RequestScoped, 325
reset(), 112
Resin, 287
@Resource, 355
resources, 388
response, 311, 313, 317
response.getWriter(), 305
response.sendRedirect(), 313
REST. *See* representation state of transfer
RESTFul Web services, 387–398
ResultSet, 26, 241–242, 443
ResultSetMetaData, 244–246
retention policy, 265, 267
reverse domain name, 13, 55
RIAs. *See* rich Internet applications
Rich Client Platform (RCP), 9
rich Internet applications (RIAs), 421
RMI. *See* Remote Method Invocation
rmiregistry, 277
rollback(), 247, 342
Root, 381
RowSet, 246–247
RPC. *See* remote procedure calls
rs, 241–242
run(), 131, 211, 215, 230, 441
RUNNABLE, 219
Runnable, 131, 210–212, 226, 441
  Executors, 226
  Thread, 222–441
RUNTIME, 265
run-time
  annotations, 270–271
  binding, 68
  errors, 70, 135
  logging, 46
RUNTIME retention policy, 267

Page 501 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

S

Scene, 429
scenes, 428
@Schedule, 372
ScheduledExecutorService, 226
scope, 20–21, 37
scriptlets, 313
SE. *See* Standard Edition
security, 278–279
security sandbox, 105
SELECT, 238, 378
selectors. *See* filters
send(), 205, 339
sendMessage(), 430
serial version unique ID, 182
Serializable, 60, 181, 444
    enum, 219
    JavaBeans, 315
    ObjectMessage, 338
    servlets, 303
    writeObject(), 177
serialization, 175–184
    byte arrays, 182–183
    classes
        variables, 179
        versioning, 182
    JVM, 182
    marker interfaces, 176
ServerSocket, 193–197
service(), 301
service_name, 279
servlets, 285, 287, 293–307
    applets, 440
    asynchronous, 306–307
    browsers, 301
    cookies, 302–303
    deploying, 296–297
    destroy(), 301
    Eclipse, 297–300
    EJB, 353
    event listeners, 305–306
    filters, 304–305
    init(), 301
    JSP, 317–318, 444
    MessageSender, 353
    metadata, 296
    presentation layer, 300
    Serializable, 303
    service(), 301
    session tracking, 302
    superclass, 301
    thin client, 294–295
    URL rewriting, 302
    writing, 295
ServletConfig, 312
ServletContext, 312
ServletContextListener, 305–306
ServletDispatcher, 402
<servlet-name>, 404
Session, 338, 339, 340, 342
    Configuration, 412
    getStore(), 204
session, 311, 317
session EJB, 286
session tracking, 302
SessionFactory, 412
Set, 146
setContentType(), 301
setDisplayValue(), 95
setDoOutput(), 188
setNull(), 243
setPageContext(), 319
setPlayerName(), 112
setPriority(), 216
setPropsAndSendEmail(), 202
setter methods, 95, 242
setText(), 94
setValueAt(), 256–257
setXXX(), 243
short, 20
showConfirmDialog(), 91
Simple Mail Transfer Protocol (SMTP), 199, 200
SimpleMail, 205
SimpleTag, 319
singleton session beans, 362, 369–370
size(), 147
sleep(), 131, 212–214, 441

Page 502 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

SMTJPSendFailedException, 201
SMTP. *See* Simple Mail Transfer Protocol
SOAP Web services, 387
Socket, 193–197
sockets, 193–197
SortedSet, 146
source code, 2, 12
SOURCE retention policy, 267
SourceForge, 412
Spring BlaseDB, 401
Spring Faces, 401
Spring Framework, 289
    bean wiring, 400
    callback methods, 399
    containers, 401
    contexts, 399
    dependency injection, 399
    Eclipse, 402–406
    Hibernate, 401
    HTML, 401–408
    HTML forms, 406–408
    MVC, 399–409
    workflows, 402–406
Spring JavaScript, 401
Spring Security, 401
Spring Web Flow, 401
springServletDispatcherservlet.xml, 404
SQL, 376, 411
sqlStatement, 266
src, 12
stack trace, 133–134
stages, 428–429
Standard Edition (SE), 3, 200, 262, 295, 375, 426
start(), 107, 211, 338, 340
startNetworkServer, 237
@Startup, 369
@Stateful, 363, 369
stateful session beans, 362, 368–369
stateless protocol, 302
stateless session beans, 362, 363–368, 446
Statement, 242–243, 443
statements, 49–51
states.txt, 170
static, 5, 440
    classes, 21, 37–38
    variables, 57, 99
Step Over, 48
stock quote program, 191–193
stop(), 107–108, 214–216, 444
stopMe, 215
Store, 204–205
stored procedures, 235
storeToXML(), 149
StreamMessage, 338
streams, 163–173
    buffered, 165–166
    byte, 164–165
    character, 166–170
    data, 171
    DBMS, 163
    File, 171–172
    I/O, 163
    network, 186
String, 51, 441
String[], 5
stubs, 273, 279
subclasses, 26, 135, 139
    abstract classes, 70
    final classes, 58
    multi-threading, 225
    RemoteException, 141
    superclasses, 440
    upcasting, 60
subscribers, 343–344
super, 34, 160, 441
superclasses, 26
    abstract classes, 70
    servlets, 301
    subclasses, 440
@SuppressWarning, 263–264
Swing, 75–88
    adapters, 97–98
    applets, 107–108, 111–122
    classes, 76
    event listeners, 440
    GUIs, 87–88, 115, 230–231
    JTable, 251–260
    layout managers, 77–88
    multi-threading, 230–233
    UI, 75, 230–231
    widgets, 86–87, 89

Page 503 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

SwingWorker, 230–233
switch, 26, 50–51, 69
SWT library, 75
synchronization, 216–219, 441
synchronized, 216–217, 225, 441
synchronizedList(), 230
synchronous calls, 336
SynchronousQueue, 229
sysinfo, 236

T

<T>, 157, 160–161
    JTable, 251
TableCellRenderer, 258
-tableChanged(), 252
TableColumn, 258
TableModel, 251
TableModelEvent, 257
TableModelListener, 252
taglib, 312
<@taglib>, 406
tags, 312–314, 318–320
@Target, 265
targets, 264–265
TCP/IP, 193
technical interviews, 435–446
templates, JSF, 326–331
TERMINATED, 219
TestGenericCollection, 158
text editor, 4–5
TextMessage, 338
Thawte, 105
thin client, 285, 294–295
thin drivers, 236
this, 34–35, 129
this(), 35
Thread, 210, 219
    join(), 223–224
    Runnable, 225, 441
    setPriority(), 216
    sleep(), 212–214
    stop(), 214–216, 444
thread-safe, 229
Thread.sleep(), 131
throw, 136, 138–139
Throwable, 135, 139, 311
throwing exceptions, 135
throws, 136, 137
tic-tac-toe, 111–122
TIMED_WAITING, 219
Timeline, 429
TimeService, 372
<title>, 106
toArray(), 160–161
TODO, 14
Topic, 338
TopicPublisher, 337, 338
TopicSubscriber, 338
toString(), 136
transactional updates, 247
transactionRequired, 266
transient, 178, 441
Transport.send(), 202
try/catch, 134, 135–136, 164, 442, 444
TYPE, 265
type erasure, 156
TypedQuery, 381
TYPE_FORWARD_ONLY, 246
TYPE_SCROLL_INSENSITIVE, 246
TYPE_SCROLL_SENSITIVE, 246

U

UI. *See* user interface
<ui:define>, 329
unbound queues, 229
unboxing, 21
unchecked errors, 135
uniform resource identifier (URI), 388
uniform resource locator (URL), 185
    rewriting, 302
unsigned applets, 104–105
upcasting, 60–61
updateData(), 266
URI. *See* uniform resource identifier
URL. *See* uniform resource locator
URL, 186, 188
UrlBasedViewResolver, 402
URLConnection, 186, 188–189

Page 504 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

urlPatterns, 295
US-ASCII, 166
UseCaches(), 188–189
user interface (UI), 76, 112–113, 124–127
    event handling, 89–102
    GUIs, 75–88, 115, 230–231
    JSF, 323
    OS, 75
    RCP, 9
    Swing, 75, 230–231
UTF-8, 166
UTF-16, 166

V

validators, 333
value, 266
values, 36–37
variables
    access level, 57
    classes, 18, 21
        constructors, 33
        initialization, 33
        serialization, 179
        static, 37–38
    declarations, 19, 20
    final, 57
    global, 148
    initialization, 19, 20
    instances, 36–37
    JSP, 311–312, 444
    properties, 22
    scope, 20–21, 37
    static, 57
        inner classes, 99
    transient, 178
VBox, 429
Vector, 41, 144–146
VeriSign, 105
versions
    class serialization, 182
    JRE, 12
void, 5, 33
volatile, 215

W

wait(), 219–222, 441
WAITING, 219
Web applications, 323–334, 439
web fragments, 289
web module, 400
web services, 285
Web Services Description Language (WSDL), 387
WEB-INF, 299, 402
WebLogic, 287
@WebServlet, 295
WebSphere, 287, 335
Welcome, 10
WHERE, 378
while, 44, 45, 46–47
widgets, 86–87, 89
width, 106
wild cards, 158–160
WindowAdapter, 98
WindowListener, 96–98
Windows installation, 3–4
workflows, 402–406
workspace, 10
World Wide Web (WWW), 185
wrappers, 21, 255
    command-line arguments, 50–51
    HTML, 114–115
write(), 166
writeByte(), 171
writeExternal(), 180
writeFloat(), 171
writeInt(), 171
writeObject(), 177, 180
WSDL. *See* Web Services Description Language
WWW. *See* World Wide Web

X

.xhtml, 323
XML, 149, 286, 392
@XmlRootElement, 388, 396

Page 505 of 506 (IPR2025-00350)
Stingray Group Inc. et al v. Hernandez-Mondragon

WILEY PUBLISHING, INC.
END-USER LICENSE AGREEMENT

READ THIS. You should carefully read these terms and conditions before opening the software packet(s) included with this book "Book". This is a license agreement "Agreement" between you and Wiley Publishing, Inc. "WPI". By opening the accompanying software packet(s), you acknowledge that you have read and accept the following terms and conditions. If you do not agree and do not want to be bound by such terms and conditions, promptly return the Book and the unopened software packet(s) to the place you obtained them for a full refund.

1. **License Grant.** WPI grants to you (either an individual or entity) a nonexclusive license to use one copy of the enclosed software program(s) (collectively, the "Software") solely for your own personal or business purposes on a single computer (whether a standard computer or a workstation component of a multi-user network). The Software is in use on a computer when it is loaded into temporary memory (RAM) or installed into permanent memory (hard disk, CD-ROM, or other storage device). WPI reserves all rights not expressly granted herein.

2. **Ownership.** WPI is the owner of all right, title, and interest, including copyright, in and to the compilation of the Software recorded on the physical packet included with this Book "Software Media". Copyright to the individual programs recorded on the Software Media is owned by the author or other authorized copyright owner of each program. Ownership of the Software and all proprietary rights relating thereto remain with WPI and its licensers.

3. **Restrictions on Use and Transfer.**

(a) You may only (i) make one copy of the Software for backup or archival purposes, or (ii) transfer the Software to a single hard disk, provided that you keep the original for backup or archival purposes. You may not (i) rent or lease the Software, (ii) copy or reproduce the Software through a LAN or other network system or through any computer subscriber system or bulletin-board system, or (iii) modify, adapt, or create derivative works based on the Software.

(b) You may not reverse engineer, decompile, or disassemble the Software. You may transfer the Software and user documentation on a permanent basis, provided that the transferee agrees to accept the terms and conditions of this Agreement and you retain no copies. If the Software is an update or has been updated, any transfer must include the most recent update and all prior versions.

4. **Restrictions on Use of Individual Programs.** You must follow the individual requirements and restrictions detailed for each individual program in the "About the CD" appendix of this Book or on the Software Media. These limitations are also contained in the individual license agreements recorded on the Software Media. These limitations may include a requirement that after using the program for a specified period of time, the user must pay a registration fee or discontinue use. By opening the Software packet(s), you agree to abide by the licenses and restrictions for these individual programs that are detailed in the "About the CD" appendix and/or on the Software Media. None of the material on this Software Media or listed in this Book may ever be redistributed, in original or modified form, for commercial purposes.

5. **Limited Warranty.**

(a) WPI warrants that the Software and Software Media are free from defects in materials and workmanship under normal use for a period of sixty (60) days from the date of purchase of this Book. If WPI receives notification within the warranty period of defects in materials or workmanship, WPI will replace the defective Software Media.

(b) WPI AND THE AUTHOR(S) OF THE BOOK DISCLAIM ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE, THE PROGRAMS, THE SOURCE CODE CONTAINED THEREIN, AND/OR THE TECHNIQUES DESCRIBED IN THIS BOOK. WPI DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE SOFTWARE WILL BE ERROR FREE.

(c) This limited warranty gives you specific legal rights, and you may have other rights that vary from jurisdiction to jurisdiction.

6. **Remedies.**

(a) WPI's entire liability and your exclusive remedy for defects in materials and workmanship shall be limited to replacement of the Software Media, which may be returned to WPI with a copy of your receipt at the following address: Software Media Fulfillment Department, Attn.: *Java Programming 24-Hour Trainer*, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, or call 1-800-762-2974. Please allow four to six weeks for delivery. This Limited Warranty is void if failure of the Software Media has resulted from accident, abuse, or misapplication. Any replacement Software Media will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

(b) In no event shall WPI or the author be liable for any damages whatsoever (including without limitation damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising from the use of or inability to use the Book or the Software, even if WPI has been advised of the possibility of such damages.

(c) Because some jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation or exclusion may not apply to you.

7. **U.S. Government Restricted Rights.** Use, duplication, or disclosure of the Software for or on behalf of the United States of America, its agencies and/or instrumentalities "U.S. Government" is subject to restrictions as stated in paragraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause of DFARS 252.227-7013, or subparagraphs (c) (1) and (2) of the Commercial Computer Software - Restricted Rights clause at FAR 52.227-19, and in similar clauses in the NASA FAR supplement, as applicable.

8. **General.** This Agreement constitutes the entire understanding of the parties and revokes and supersedes all prior agreements, oral or written, between them and may not be modified or amended except in a writing signed by both parties hereto that specifically refers to this Agreement. This Agreement shall take precedence over any other documents that may be in conflict herewith. If any one or more provisions contained in this Agreement are held by any court or tribunal to be invalid, illegal, or otherwise unenforceable, each and every other provision shall remain in full force and effect.