public class Name {

private String fName;

private String mName;

private String lName;

public Name(String fName, String mName, String lName) {

setfName(fName);

setmName(mName);

setlName(lName);

}

public String getfName() {

return fName;

}

public void setfName(String fName) {

this.fName = fName;

}

public String getmName() {

return mName;

}

public void setmName(String mName) {

this.mName = mName;

}

public String getlName() {

return lName;

}

public void setlName(String lName) {

this.lName = lName;

}

*@Override*

public String toString() {

return "Name: " + getfName() + " " + getmName().charAt(0) + ". " + getlName();

}

}

public class Person extends Name {

private int age;

public Person(String fName, String mName, String lName, int age) {

super(fName, mName, lName);

setAge(age);

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

*@Override*

public String toString() {

return super.toString() + "\n" + "Age: " + getAge();

}

}public interface Action {

public abstract String doThis();

public abstract String sport();

}

public abstract class Athlete extends Person implements Action {

private String team;

private String position;

public Athlete(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age);

setTeam(team);

setPosition(position);

}

public abstract int ranking();

public String getTeam() {

return team;

}

public void setTeam(String team) {

this.team = team;

}

public String getPosition() {

return position;

}

public void setPosition(String position) {

this.position = position;

}

public String doThis() {

return "I am an Athlete that play major sports";

}

*@Override*

public String toString() {

return super.toString() + "\n" + "Team: " + getTeam() + "\n" + "Position: " + getPosition();

}

public class Baseball extends Athlete implements Action{

Random rand = new Random();

public Baseball(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age, team, position);

}

*@Override*

public String sport() {

return "Baseball";

}

*@Override*

public int ranking() {

return rand.nextInt(11) + 1;

}

*@Override*

public String doThis() {

return "I hit home Runs!";

}

*@Override*

public boolean equals(Object otherObj) {

if(!(otherObj instanceof Baseball))

return false;

else

//returns "true" if test is true

return ((this.ranking() == ((Baseball)otherObj).ranking())

&& (this.ranking() == ((Baseball) otherObj).ranking()));

}

*@Override*

public String toString() {

return super.toString() + "\n" + "Ranking: " + ranking() + "\n" + "SPORT: " + sport();

}

public class Hockey extends Athlete implements Action{

public Hockey(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age, team, position);

}

*@Override*

public String sport() {

return "Hockey";

}

*@Override*

public String doThis() {

return "I play on ICE";

}

*@Override*

public int ranking() {

Random rand = new Random();

return rand.nextInt(11) + 1;

}

*@Override*

public boolean equals(Object otherObj) {

if(!(otherObj instanceof Hockey))

return false;

else

//returns "true" if test is true

return ((this.ranking() == ((Hockey)otherObj).ranking())

&& (this.ranking() == ((Hockey) otherObj).ranking()));

}

*@Override*

public String toString() {

String msg = "Name: " + super.getfName() + " " + super.getmName().charAt(0) + ". " + super.getlName() + "\n";

String msg2 = "Age: " + super.getAge() + "\n" + "Team: " + super.getTeam() + "\n" + "IS A: " + super.getPosition() + "\n" + "Ranking: "

+ ranking() + "\n" + "SPORT: " + sport();

return msg + msg2;

}

public class Football extends Athlete implements Action{

Random rand = new Random();

private int tackles;

private int yardsRan;

private int yardsThrown;

public Football(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age, team, position);

}

public int getTackles() {

return tackles;

}

public void setTackles(int tackles) {

this.tackles = rand.nextInt(tackles);

}

public int getYardsRan() {

return yardsRan;

}

public void setYardsRan(int yardsRan) {

this.yardsRan = rand.nextInt(yardsRan);

}

public int getYardsThrown() {

return yardsThrown;

}

public void setYardsThrown(int yardsThrown) {

this.yardsThrown = rand.nextInt(yardsThrown);

}

*@Override*

public int ranking() {

return rand.nextInt(11) + 1;

}

*@Override*

public String sport() {

return "Football";

}

*@Override*

public boolean equals(Object otherObj) {

if(!(otherObj instanceof Football))

return false;

else

//returns "true" if test is true

return ((this.ranking() == ((Football)otherObj).ranking())

&& (this.ranking() == ((Football) otherObj).ranking()));

}

*@Override*

public String doThis() {

return "I score TOUCHDOWNS!";

}

*@Override*

public String toString() {

return super.toString() + "\n" + "Ranking: " + ranking() + "\n" + "SPORT: " + sport();

}

public class Golfer extends Athlete implements Action {

public Golfer(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age, team, position);

}

public String sport() {

return "GOLF";

}

public String doThis() {

return "I got a Hole in one!";

}

*@Override*

public int ranking() {

Random rand = new Random();

return rand.nextInt(11) + 1;

}

*@Override*

public boolean equals(Object otherObj) {

if(!(otherObj instanceof Golfer))

return false;

else

//returns "true" if test is true

return ((this.ranking() == ((Golfer)otherObj).ranking())

&& (this.ranking() == ((Golfer) otherObj).ranking()));

}

*@Override*

public String toString() {

String msg = "Name: " + super.getfName() + " " + super.getmName().charAt(0) + " " + super.getlName() + "\n";

String msg2 = "Age: " + super.getAge() + "\n" + "Tournment: " + super.getTeam() + "\n" + "Sponsor: " + super.getPosition() + "\n" + "Ranking: "

+ ranking() + "\n"+ "SPORT: " + sport();

return msg + msg2;

}

public class Hockey extends Athlete implements Action{

public Hockey(String fName, String mName, String lName, int age, String team, String position) {

super(fName, mName, lName, age, team, position);

}

*@Override*

public String sport() {

return "Hockey";

}

*@Override*

public String doThis() {

return "I play on ICE";

}

*@Override*

public int ranking() {

Random rand = new Random();

return rand.nextInt(11) + 1;

}

*@Override*

public boolean equals(Object otherObj) {

if(!(otherObj instanceof Hockey))

return false;

else

//returns "true" if test is true

return ((this.ranking() == ((Hockey)otherObj).ranking())

&& (this.ranking() == ((Hockey) otherObj).ranking()));

}

*@Override*

public String toString() {

String msg = "Name: " + super.getfName() + " " + super.getmName().charAt(0) + ". " + super.getlName() + "\n";

String msg2 = "Age: " + super.getAge() + "\n" + "Team: " + super.getTeam() + "\n" + "IS A: " + super.getPosition() + "\n" + "Ranking: "

+ ranking() + "\n" + "SPORT: " + sport();

return msg + msg2;

}