1. **public** **class** Animal {

**private** **static** Animal *a* = **null**;

**public** String name;

**private** **void** Animal()

{

name = "Lion";

}

**public** **static** Animal getA() {

**if**(*a* == **null**)

{

*a* = **new** Animal();

}

**return** *a*;

}

}

**public** **class** Wild{

**public** **static** **void** main(String[] args)

{

Animal s = Animal.*getA*();

s.name = (s.name).toUpperCase();

System.***out***.println("String from s :"+ s.name);

}

}

1. **public** **abstract** **class** employee {

String name;

**int** age;

String designation;

**int** basic\_pay;

**int** incentives;

**int** over\_time;

**public** employee(String name, **int** age, String designation, **int** basic\_pay, **int** incentives, **int** over\_time){

**this**.name = name;

**this**.age = age;

**this**.designation = designation;

**this**.basic\_pay = basic\_pay;

**this**.incentives = incentives;

**this**.over\_time = over\_time;

}

**public** **abstract** **void** salary();

}

**public** **class** Manager **extends** employee {

**public** Manager(String name, **int** age, String designation, **int** basic\_pay, **int** incentives, **int** over\_time) {

**super**(name, age, designation, basic\_pay, incentives, over\_time);

}

**public** **void** salary() {

**int** salary = basic\_pay + incentives;

System.***out***.println(name);

System.***out***.println(age);

System.***out***.println(designation);

System.***out***.print("total salary - " + salary);

}

}

**public** **class** labour **extends** employee {

**public** labour(String name, **int** age, String designation, **int** basic\_pay, **int** incentives, **int** over\_time) {

**super**(name, age, designation, basic\_pay, incentives, over\_time);

}

**public** **void** salary() {

**int** salary = basic\_pay + over\_time;

System.***out***.println("\n\n" + name);

System.***out***.println(age);

System.***out***.println(designation);

System.***out***.print("total salary - " + salary);

}

}

**public** **class** Organisation {

**public** **static** **void** main(String[] args)

{

Manager m = **new** Manager("Ram", 37, "Manager", 120000, 40000, 0);

m.salary();

labour l = **new** labour("Lakshman", 25, "Labour", 12000, 0, 10000);

l.salary();

}

}

1. **class** account{

String name;

**int** bank\_Id;

**int** savings;

account(String name, **int** bank\_Id, **int** savings){

**this**.name = name;

**this**.bank\_Id = bank\_Id;

**this**.savings = savings;

}

}

**class** savings **extends** account{

**private** **int** fixed\_deposit = 100000;

savings(String name, **int** bank\_Id, **int** savings) {

**super**(name, bank\_Id, savings);

**this**.savings = **this**.savings + fixed\_deposit;

}

**public** **void** savings(){

System.***out***.println(savings + " is the amount in savings account of " + name);

System.***out***.println(fixed\_deposit + " is the amount of fixed deposit\n");

}

}

**class** current **extends** account{

**private** **int** cash\_credits = 75000;

current(String name, **int** bank\_Id, **int** savings) {

**super**(name, bank\_Id, savings);

**this**.savings = **this**.savings + cash\_credits;

}

**public** **void** current() {

System.***out***.println(savings + " is the amount in current account of " + name);

System.***out***.println(cash\_credits+ " is the cash credits\n");

}

}

**public** **class** bank {

**public** **static** **void** main(String[] args)

{

savings a = **new** savings("Ram", 112345, 400000);

current b = **new** current("Raghu", 334567, 700000);

a.savings();

b.current();

**int** total = a.savings+b.savings;

System.***out***.println(total + " is the total amount in bank");

}

}

1. **a. public** **class** abstract\_c {

**public** **abstract** **void** move();

}

Error: abstract class to define abstract methods

**b. public** abstract **class** abstract\_c {

**public** **abstract** **void** move();

}

**public** **class** abst {

**public** **static** **void** main(String[] args) {

abstract\_c A = new abstract\_c();

creature C = **new** creature();

C.move();

}

}

Exception in thread "main" java.lang.Error: Unresolved compilation problem:

Cannot instantiate the type abstract\_c

at abst.main(abst.java:4)

**c. public** abstract **class** abstract\_c {

**public** **abstract** **void** move();

}

**public** **class** abst {

**public** **static** **void** main(String[] args) {

//abstract\_c A = new abstract\_c();

creature C = **new** creature();

C.move();

}

}

**public** **class** creature **extends** abstract\_c{

/\***public** **void** move() {

System.***out***.println("every creatures int the world moves.");

}\*/

}

Exception in thread "main" java.lang.Error: Unresolved compilation problem:

The type creature must implement the inherited abstract method abstract\_c.move()

at creature.move(creature.java:2)

at abst.main(abst.java:6)

**d.** **After changing the public into private in abstract\_c:**

**private** **abstract** **class** abstract\_c {

**public** **abstract** **void** move();

}

**public** **class** abst {

**public** **static** **void** main(String[] args) {

//abstract\_c A = new abstract\_c();

creature C = **new** creature();

C.move();

}

}

**public** **class** creature **extends** abstract\_c{

**public** **void** move() {

System.***out***.println("every creatures int the world moves.");

}

}

Exception in thread "main" java.lang.Error: Unresolved compilation problem:

Implicit super constructor abstract\_c() is not visible for default constructor. Must define an explicit constructor

at creature.<init>(creature.java:2)

at abst.main(abst.java:5)

**e. After adding the final keyword in the abstract class in abstract\_c:**

**public** **final** **abstract** **class** abstract\_c {

**public** **abstract** **void** move();

}

**public** **class** abst {

**public** **static** **void** main(String[] args) {

//abstract\_c A = new abstract\_c();

creature C = **new** creature();

C.move();

}

}

**public** **class** creature **extends** abstract\_c{

**public** **void** move() {

System.***out***.println("every creatures int the world moves.");

}

}

Exception in thread "main" java.lang.Error: Unresolved compilation problem:

The class abstract\_c can be either abstract or final, not both

at abstract\_c.<init>(abstract\_c.java:2)

at creature.<init>(creature.java:2)

at abst.main(abst.java:5)

**f. public** **abstract** **class** abstract\_c {

**public** **abstract** **void** move(){

}

}

**public** **class** abst {

**public** **static** **void** main(String[] args) {

//abstract\_c A = new abstract\_c();

creature C = **new** creature();

C.move();

}

}

**public** **class** creature **extends** abstract\_c{

**public** **void** move() {

System.***out***.println("every creatures int the world moves.");

}

}

1. **import** java.util.Scanner;

**public** **abstract** **class** shape {

**public** **abstract** **void** draw();

}

**class** line **extends** shape{

Scanner scn = **new** Scanner(System.***in***);

**public** **void** draw() {

System.***out***.println("Enter the lenght of the line:");

**int** n = scn.nextInt();

**for** (**int** i=0; i<n; i++) {

System.***out***.print("\*");

}

System.***out***.println("\n\n");

}

}

**class** rectangle **extends** shape{

Scanner scn = **new** Scanner(System.***in***);

**public** **void** draw() {

System.***out***.println("Enter the length of the rectangle:");

**int** n = scn.nextInt(); //i

System.***out***.println("Enter the width of the rectangle:");

**int** m = scn.nextInt(); //j

**for** (**int** i=0; i<n; i++) {

**for**(**int** j=0; j<m; j++) {

**if**(i==0 || i== n-1 || j==0 || j== m-1)

System.***out***.print("\*");

**else**

System.***out***.print(" ");

}

System.***out***.println();

}

}

}

**class** square **extends** shape{

Scanner scn = **new** Scanner(System.***in***);

**public** **void** draw() {

System.***out***.println("Enter the length of the square:");

**int** n = scn.nextInt(); //i

System.***out***.println("Enter the width of the sqaure:");

**int** m = scn.nextInt(); //j

**for** (**int** i=0; i<n; i++) {

**for**(**int** j=0; j<m; j++) {

**if**(i==0 || i== n-1 || j==0 || j== m-1)

System.***out***.print("\* ");

**else**

System.***out***.print(" ");

}

System.***out***.println(" ");

}

}

}

**class** cube **extends** shape{

Scanner scn = **new** Scanner(System.***in***);

**public** **void** draw() {

System.***out***.println("Enter the length of the cube:");

**int** n = scn.nextInt(); //i

System.***out***.println("Enter the width of the cube:");

**int** m = scn.nextInt(); //j

**for** (**int** i=0; i<n; i++) {

**for**(**int** j=0; j<m; j++) {

**if**((i==0 && j==0) || (i== n-1 && j== m-1))

System.***out***.print(" ");

**else**

System.***out***.print("\* ");

}

System.***out***.println(" ");

}

}

}

**public** **class** shape\_2 {

**public** **static** **void** main(String[] args) {

line l = **new** line();

rectangle r = **new** rectangle();

square s = **new** square();

cube c = **new** cube();

l.draw();

r.draw();

s.draw();

c.draw();

}

}

1. **public** **abstract** **class** persistence {

**public** **abstract** **void** persist();

}

**class** FilePersistence **extends** persistence{

**public** **void** persist() {

System.***out***.println("Data is stored in file");

}

}

**class** DatabasePersistence **extends** persistence{

**public** **void** persist() {

System.***out***.println("Data is stored in database");

}

}

**import** java.util.Scanner;

**public** **class** persistence\_2 {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the Persistence Type you want your data to be saved in: ");

String persistence = sc.nextLine();

**if** (persistence.equals("File Persistence"))

{

FilePersistence fp = **new** FilePersistence() {};

fp.persist();

}

**if** (persistence.equals("Database Persistence"))

{

DatabasePersistence db = **new** DatabasePersistence() {};

db.persist();

}

}

}

1. **public** **abstract** **class** Dessert\_Items {

**public** **abstract** **int** getCost(**int** countity);

}

**public** **class** Cookie **extends** Dessert\_Items{

//@Override

**public** **int** getCost(**int** countity) {

**return** countity\*60;

}

}

**public** **class** Candy **extends** Dessert\_Items {

@Override

**public** **int** getCost(**int** countity) {

// **TODO** Auto-generated method stub

**return** countity\*70;

}

}

**public** **class** IceCream **extends** Dessert\_Items {

@Override

**public** **int** getCost(**int** countity) {

// **TODO** Auto-generated method stub

**return** countity\*1;

}

}

**import** java.util.Scanner;

**public** **class** Bakery{

**public** **static** **void** main(String[] args) {

Dessert\_Items dessertItem=**null**;

String item1="candy";

String item2="cookie";

String item3="icecreams";

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println(" enter your role");

String role=sc.next();

**if**(role.equalsIgnoreCase("owner"))

{

System.***out***.println("welcome owner add items");

item1=sc.next();

**if**(item1.equalsIgnoreCase("icecream")||item1.equalsIgnoreCase("candy")||item1.equalsIgnoreCase("cookie"))

{

System.***out***.println(item1+" has added");

}

**else**

{

System.***out***.println("you are added invalid item");

}

item2=sc.next();

**if**(item2.equalsIgnoreCase("icecream")||item2.equalsIgnoreCase("candy")||item2.equalsIgnoreCase("cookie"))

{

System.***out***.println(item2+" has added");

}

**else**

{

System.***out***.println("you are added invalid item");

}

item3=sc.next();

**if**(item3.equalsIgnoreCase("icecream")||item3.equalsIgnoreCase("candy")||item3.equalsIgnoreCase("cookie"))

{

System.***out***.println(item3+" has added");

}

**else**

{

System.***out***.println("you are added invalid item");

}

}

**else** **if**(role.equalsIgnoreCase("customer"))

{

**int** cost1=0,cost2=0,cost3=0;

System.***out***.println("these are available items");

System.***out***.println(item1+" "+item2+" "+item3);

**for**(**int** i=1;i<3;i++)

{

System.***out***.println("enter "+i+" item");

String customerSelectedItem=sc.next();

System.***out***.println("enter quantity");

**int** quantity=sc.nextInt();

**if**(customerSelectedItem.equalsIgnoreCase("candy"))

{

dessertItem=**new** Candy();

**int** price=dessertItem.getCost(quantity);

System.***out***.println("candy quantity "+quantity+" price is "+price);

}

**else** **if**(customerSelectedItem.equalsIgnoreCase("cokkie"))

{

dessertItem=**new** Cookie();

**int** price=dessertItem.getCost(quantity);

System.***out***.println("cokkie quantity "+quantity+" price is "+price);

}

**else** **if**(customerSelectedItem.equalsIgnoreCase("icecream"))

{

dessertItem=**new** IceCream();

**int** price=dessertItem.getCost(quantity);

System.***out***.println("icecream quantity "+quantity+" price "+price);

}

}

}

}

}