package class19;Intellji

public class Task1 {

/\*

Write a Student class that have instance variables name and address.

Create a constructor that will initialize those variables.

Print name & address of given student using displayInfo method.

\*/

public static void main(String[] args) {

Student josh=new Student("Naughty Josh","Earth");

josh.displayInfo();

}

}

class Student{

String name;

String address;

public Student(String name,String address){

this.name=name;

this.address=address;

}

void displayInfo(){

System.out.println("Name "+name+" Address "+address);

}

}

package class19;

public class Animal {

String name;

String breed;

String color;

public Animal(String name, String breed, String color) {

this.name = name;

this.breed = breed;

this.color = color;

}

public void printInfo(){

System.out.println("Name "+name+" Breed "+breed+" Color "+color);

}

}

============================================================================================

package class19;

public class Bird {

String name;

String color;

int age;

double weight;

public Bird(String name,String color,int age,double weight){

this.name=name;

this.color=color;

this.age=age;

this.weight=weight;

}

void printInfo(){

System.out.println("Name "+name +" Color "+color+" Age "+age+

" Weight "+weight);

}

}

class Test{

public static void main(String[] args) {

Parrot parrot=new Parrot("Zeki","Green",

2,1);

parrot.printInfo();

Crow crow=new Crow("crow","black",3,1.5);

crow.printInfo();

Sparrow sparrow=new Sparrow("Sparrow","White",3,1.5);

sparrow.printInfo();

}

}

package class19;

class Vehicle {

int noOfWheel;

int topSpeed;

public void printTopSpeed(){

System.out.println(topSpeed);

}

}

public class Car extends Vehicle{

String model;

String make;

int year;

}

class Tesla extends Car{

void printInfo(){

System.out.println("I am a Electrical Car");

}

}

class Toyota extends Car{

void printInfo(){

System.out.println("I am a Combustion Engine Car");

}

}

/\*class CustomCar extends Toyota,Tesla{

not allowed because of diamond problem

}\*/

class Tester{

public static void main(String[] args) {

Toyota toyota=new Toyota();

toyota.printTopSpeed();

}

}

package class19;

class Parrot extends Bird{

Parrot(String name,String color,int age,double weight){

super(name, color, age, weight);

}

}

class Crow extends Bird{

Crow(String name,String color,int age,double weight ){

super(name, color, age, weight);

}

}

class Sparrow extends Bird{

Sparrow(String name,String color,int age,double weight){

super(name, color, age, weight);

}

}

package class19;

class Parrot extends Bird{

Parrot(String name,String color,int age,double weight){

super(name, color, age, weight);

}

}

class Crow extends Bird{

Crow(String name,String color,int age,double weight ){

super(name, color, age, weight);

}

}

class Sparrow extends Bird{

Sparrow(String name,String color,int age,double weight){

super(name, color, age, weight);

}

}

package class19;

public class Dog extends Animal{

public Dog(String name,String breed,String color){

super(name, breed, color);

}

}

class Cat extends Animal{

Cat(String name,String breed,String color){

super(name,breed,color);

}

}

class TestAnimal{

public static void main(String[] args) {

Dog dog=new Dog("jacky","German","Black");

dog.printInfo();

}

}

package class19;

public class Furniture {

String color="green";

}

class Chair extends Furniture{

String color="Red";

void printColor(){

String color="Black";

System.out.println(color); //prints the local variable

System.out.println(this.color); // prints the instance variable from same clas

System.out.println(super.color); // prints the instance variable of parent

//Break till 1:47

}

}

class TestChair{

public static void main(String[] args) {

Chair chair=new Chair();

chair.printColor();

}

}

"this" keyword:

"this" is a keyword like "class", "void". it is used in many places. For example

if we have same name instance and a local variable we can use "this" to

explicitly refer to instance variable. We can use "this()" to call a constructor inside another

constructor.

why should we learn about "this" keyword:

1)it allows us to have same name local and instance variables.

2) if we have same code in two constructors we can reuse that code from one constructor

inside another constructor by writing "this()".

Inheritance:

Inheritance is a mechanism in java through which we can use the fields/methods of

one class inside another class.

There are 4 types of inheritance.

1) Single inheritance:

In single inheritance we have only one parent and one child class.

Employee->Person.

2) Multilevel Inheritance:

In Multilevel inheritance we have inheritance in multiple levels like we have a grandfather-> father->son->grand son

3) Hierarchical inheritance:

one parent class and multiple direct child classes.

4) Multiple inheritance:

one child class can directly inherit from multiple parent classes. it is not supported by java because of diamond problem.

Why should we learn inheritance?

It allows to reduce the duplicate code.

what is the syntax?

we can use this feature by writing extends keyword when declaring a class.

package class19;

public class Task1 {

/\*

Write a Student class that have instance variables name and address.

Create a constructor that will initialize those variables.

Print name & address of given student using displayInfo method.

\*/

public static void main(String[] args) {

Student josh=new Student("Naughty Josh","Earth");

josh.displayInfo();

}

}

class Student{

String name;

String address;

public Student(String name,String address){

this.name=name;

this.address=address;

}

void displayInfo(){

System.out.println("Name "+name+" Address "+address);

}

}

package class19;

public class Task1 {

/\*

Write a Student class that have instance variables name and address.

Create a constructor that will initialize those variables.

Print name & address of given student using displayInfo method.

\*/

public static void main(String[] args) {

Student josh=new Student("Naughty Josh","Earth");

josh.displayInfo();

}

}

class Student{

String name;

String address;

public Student(String name,String address){

this.name=name;

this.address=address;

}

void displayInfo(){

System.out.println("Name "+name+" Address "+address);

}

}

package class19;

public class Task2 {

/\*

Write Book class that will have instance variables and 2 Constructors. While creating an object make sure:

Instance variables are being initialized

Both Constructors are being executed

\*/

public static void main(String[] args) {

Book headFirstJava=new Book("HeadIFirstJava","Kathy",12,13345);

new Book("HeadIFirstJava","Kathy",12);

//Break till 11:45

}

}

class Book{

String name;

String author;

int noOfTotalCh;

int ISBN;

public Book(String name,String author,int noOfTotalCh,int ISBN){

/\* this.name=name;

this.author=author;

this.noOfTotalCh=noOfTotalCh;\*/

this(name, author, noOfTotalCh);

this.ISBN=ISBN;

}

public Book(String name,String author,int noOfTotalCh){

this.name=name;

this.author=author;

this.noOfTotalCh=noOfTotalCh;

}

}