package class20;inteliji

import java.util.Scanner;

public class MethodOverloading {

int number;

double dNumber;

MethodOverloading() {

}

MethodOverloading(double dNumber) {

this.dNumber = dNumber;

}

MethodOverloading(int number) {

this.number = number;

}

static void add(int num1, int num2) {

System.out.println(num1 + num2);

}

static void add(double num1, double num2) {

System.out.println(num1 + num2);

}

static void add(double num1, double num2, double num3) {

System.out.println(num1 + num2 + num3);

}

public static void main(String[] args) {

add(20, 20);

new MethodOverloading(45);

new Scanner(System.in);

}

}

package class20;

public class MethodOverloadingD2 {

private static MethodOverloading md;

void F1(String name, int number){

System.out.println("1");

}

// By changing the number of parameters

void F1(String name,int number,int number2){

System.out.println("2");

}

//By changing the data types

void F1(int num1,int number){

System.out.println("3");

}

void F1(int number,String name){

System.out.println("4");

}

public static void main(String[] args) {

MethodOverloadingD2 md=new MethodOverloadingD2();

md.F1("Taras",20);

}

}

package class20;

public class Repl139 {

public static void main(String[] args) {

char c='B';

System.out.println((int)c);

for (int i = 65; i <120 ; i++) {

System.out.println((char)i);

}

if('A'>'B'){

System.out.println("its a false");

}

}

}

What is inheritance?

It's a mechanism through which a class can inherit the properties and methods of another class.

Why we should learn inheritance?

It helps use avoid code duplication.

It helps us save space.

It helps us improve the execution speed.

What is the syntax?

We use the extends keyword and the name of the class from which we want to inherit the fields

and methods while declaring the class.

Types of inheritance:

There can be 4 types of inheritance but java only supports 3 of them with classes.

1) Single.

2) Multilevel.

3) Hierarchical.

4) Multiple(it is not supported by classes in java).

Single inheritance:

When we have only one parent and one child classes in inheritance it is called single inheritance.

not widely used.

Multilevel Inheritance:

One top level and one base class and many intermediate classes. For Example A<B<C<D.

Hierarchical Inheritance:

When we have one parent class and multiple child classes it's called Hierarchical Inheritance.

Multiple Inheritance:

One child and multiple parent classes. it is not supported by classes in java. It creates the diamond problem.

Super keyword:

We use the super keyword to explicitly call the variable from the parent class. When we have

same name instance variable in child class and parent class.

We can also use the super keywrod to call the constructor of the parent class.

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

MethodOverloading:

Method overloading allows us to create multiple methods in a class with same name.

Why we should create multiple methods with same name?

It gives the end user flexibility.

The end user can call the same method with different parameters to get same or different results.

How do we overload a method?

1) By passing different number of parameters.

2) By changing the datatype.

3) By changing the sequence of parameters

package class20;

public class Task1 {

/\*Write program for multilevel inheritance

where class C inherits from class B and Class B inherits from Class A.

\*/

public static void main(String[] args) {

// C c=new C();

B b=new B();

}

}

class A{

A(){

System.out.println("A");

}

}

class B extends A{

B(){

System.out.println("B");

}

}

class C extends B{

C(){

System.out.println("C");

}

}

package class20;

public class Task2 {

/\*

HW2: Write program to inherit class Z that has method printF

which is static and call or reuse that method into class W

\*/

public static void main(String[] args) {

Z.printF();

W.printF();

}

}

class Z{

static void printF(){

System.out.println("F");

}

}

class W extends Z{

}

package class20;

public class Task3 {

/\*

Write a Java program called Teacher.

Identify features and behaviour of that Class. Create 3 subclasses MathTeacher,

ChemistryTeacher and PianoTeacher that would have it their own features and behaviour.

Test all 4 classes.

\*/

}

class Teacher {

protected String name;

protected int age;

protected int yearsOfExperience;

Teacher(String name, int age, int yearsOfExperience) {

this.name = name;

this.age = age;

this.yearsOfExperience = yearsOfExperience;

}

// Break till 8:55

void teach() {

System.out.println(name + " is teaching");

}

}

class MathTeacher extends Teacher{

void teachTheClass(){

System.out.println("I will teach the class even the world is going to end right after my class");

}

MathTeacher(String name,int age,int noOfYear){

super(name,age,noOfYear);

}

}

class ChemistryTeacher extends Teacher{

boolean teachOrganic;

ChemistryTeacher(String name, int age, int yearsOfExperience) {

super(name, age, yearsOfExperience);

}

void teachChem(){

System.out.println(name+" teaching Chemistry");

}

}

class PianoTeacher extends Teacher{

String pianoType;

PianoTeacher(String name, int age, int yearsOfExperience,String pianoType) {

super(name, age, yearsOfExperience);

this.pianoType=pianoType;

}

}

package class20;

public class Task4 {

/\*

Write program: userClass that has a constructor that initializes instance variable

name and mobile number. Create a subclass userInfo that will have user address variable

and it also being initialized through constructor call. Print users name, mobile number

and address in userDetails method. Test your code.

\*/

}

package class20;

public class TeacherTester {

public static void main(String[] args) {

MathTeacher mathTeacher=

new MathTeacher("Gulcan",18,5);

mathTeacher.teach();

PianoTeacher pt=new PianoTeacher(

"James",

25,

6,"Grand"

);

pt.teach();

}

}

<https://replit.com/teams/join/jszkwhnsmnghidzuozheurwuwgkarfxq-SyntaxSDETBatch14>