**SETCION 7-2:**

Parameters and Overloading Methods Practice Activities Lesson Objectives:

• Use access modifiers

• Pass objects to methods

• Return objects from methods

• Use variable argument methods

• Overload constructors

• Overload methods

• Write a class with specified arrays, constructors, and methods

1. **Program:**

**package** dffg;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

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

Fish fish1 = **new** Fish();

fish1.setTypeOfFish("Goldfish");

fish1.setFriendliness(8);

Fish fish2 = **new** Fish("Betta", 5);

System.***out***.println("Fish 1: " + fish1.getTypeOfFish() + ", Friendliness: " + fish1.getFriendliness());

System.***out***.println("Fish 2: " + fish2.getTypeOfFish() + ", Friendliness: " + fish2.getFriendliness());

}

}

**Output:**

****

1. **Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

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

Fish fish1 = **new** Fish();

Fish fish2 = **new** Fish("Betta", 5);

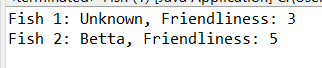
System.***out***.println("Fish 1: " + fish1.getTypeOfFish() + ", Friendliness: " + fish1.getFriendliness());

System.***out***.println("Fish 2: " + fish2.getTypeOfFish() + ", Friendliness: " + fish2.getFriendliness());

}

}

**Output:**



1. **Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

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

Fish fish1 = **new** Fish();

Fish fish2 = **new** Fish("Betta", 5);

Fish fish3 = **new** Fish("Guppy", 7);

System.***out***.println("Fish 1: " + fish1.getTypeOfFish() + ", Friendliness: " + fish1.getFriendliness());

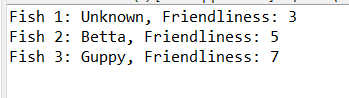
System.***out***.println("Fish 2: " + fish2.getTypeOfFish() + ", Friendliness: " + fish2.getFriendliness());

System.***out***.println("Fish 3: " + fish3.getTypeOfFish() + ", Friendliness: " + fish3.getFriendliness());

}

}

**Output:**

****

1. **Program:**

**Default Constructor**:  
public Fish() {

this.typeOfFish = "Unknown";

this.friendliness = 3;

}

**Parameterized Constructor**:

public Fish(String typeOfFish, int friendliness) {

this.typeOfFish = typeOfFish;

this.friendliness = friendliness;

}

**Additional Parameterized Constructor**:

public Fish(String t, int f) {

this.typeOfFish = t;

this.friendliness = f;

}

**Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

**public** **int** getFriendlinessLevel() {

**return** friendliness;

}

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

Fish fish1 = **new** Fish();

Fish fish2 = **new** Fish("Betta", 5);

Fish fish3 = **new** Fish("Guppy", 7);

System.***out***.println("Fish 1 Friendliness: " + fish1.getFriendlinessLevel());

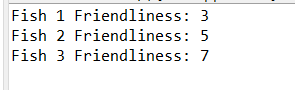
System.***out***.println("Fish 2 Friendliness: " + fish2.getFriendlinessLevel());

System.***out***.println("Fish 3 Friendliness: " + fish3.getFriendlinessLevel());

}

}

**Output:**

****



**Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

**public** **int** getFriendlinessLevel() {

**return** friendliness;

}

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

Fish fish1 = **new** Fish("AngelFish", 5);

Fish fish2 = **new** Fish("Guppy", 3);

System.***out***.println("Fish 1: Type – " + fish1.getTypeOfFish() + ", Friendliness level – " + fish1.getFriendlinessLevel());

System.***out***.println("Fish 2: Type – " + fish2.getTypeOfFish() + ", Friendliness level – " + fish2.getFriendlinessLevel());

}

}

**Output:**

****

**Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

**public** **int** getFriendlinessLevel() {

**return** friendliness;

}

**public** **static** Fish nicestFish(Fish fish1, Fish fish2) {

**if** (fish1.getFriendlinessLevel() > fish2.getFriendlinessLevel()) {

**return** fish1;

} **else** {

**return** fish2;

}

}

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

Fish fish1 = **new** Fish("AngelFish", 5);

Fish fish2 = **new** Fish("Guppy", 3);

Fish nicerFish = Fish.*nicestFish*(fish1, fish2);

System.***out***.println("The nicest fish is of type – " + nicerFish.getTypeOfFish() + " with a friendliness level of – " + nicerFish.getFriendlinessLevel());

}

}

**Output:**

****

**8.**

**Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

**public** **int** getFriendlinessLevel() {

**return** friendliness;

}

**public** **static** Fish nicestFish(Fish... fishArray) {

**if** (fishArray == **null** || fishArray.length == 0) {

**throw** **new** IllegalArgumentException("No fish provided");

}

Fish temp = fishArray[0];

**for** (Fish fish : fishArray) {

**if** (fish.getFriendlinessLevel() > temp.getFriendlinessLevel()) {

temp = fish;

}

}

**return** temp;

}

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

Fish fish1 = **new** Fish("AngelFish", 5);

Fish fish2 = **new** Fish("Guppy", 3);

Fish fish3 = **new** Fish("Betta", 4);

Fish nicest = Fish.*nicestFish*(fish1, fish2, fish3);

System.***out***.println("The nicest fish is of type – " + nicest.getTypeOfFish() + " with a friendliness level of – " + nicest.getFriendlinessLevel());

}

}

**Output:**

****

**9.**

**Program:**

**package** nnn;

**public** **class** Fish {

**private** String typeOfFish;

**private** **int** friendliness;

**public** Fish() {

**this**.typeOfFish = "Unknown";

**this**.friendliness = 3;

}

**public** Fish(String typeOfFish, **int** friendliness) {

**this**.typeOfFish = typeOfFish;

**this**.friendliness = friendliness;

}

**public** Fish(String t, **int** f) {

**this**.typeOfFish = t;

**this**.friendliness = f;

}

**public** String getTypeOfFish() {

**return** typeOfFish;

}

**public** **void** setTypeOfFish(String typeOfFish) {

**this**.typeOfFish = typeOfFish;

}

**public** **int** getFriendliness() {

**return** friendliness;

}

**public** **void** setFriendliness(**int** friendliness) {

**this**.friendliness = friendliness;

}

**public** **int** getFriendlinessLevel() {

**return** friendliness;

}

**public** **static** Fish nicestFish(Fish... fishArray) {

**if** (fishArray == **null** || fishArray.length == 0) {

**throw** **new** IllegalArgumentException("No fish provided");

}

Fish temp = fishArray[0];

**for** (Fish fish : fishArray) {

**if** (fish.getFriendlinessLevel() > temp.getFriendlinessLevel()) {

temp = fish;

}

}

**return** temp;

}

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

Fish fish1 = **new** Fish("AngelFish", 6);

Fish fish2 = **new** Fish("Guppy", 3);

Fish nicest = Fish.*nicestFish*(fish1, fish2);

System.***out***.println("The nicest fish is of type – " + nicest.getTypeOfFish() + " with a friendliness level of – " + nicest.getFriendlinessLevel());

}

}

**Output:**

****

**10.**

**Program:**

public class Employee {

private String name;

private String address;

private double salary;

private String phoneNumber;

public Employee(String name, String address, double salary, String phoneNumber) {

this.name = name;

this.address = address;

this.salary = salary;

this.phoneNumber = phoneNumber;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public double getSalary() {

return salary;

}

public void setSalary(double salary) {

this.salary = salary;

}

public String getPhoneNumber() {

return phoneNumber;

}

public void setPhoneNumber(String phoneNumber) {

this.phoneNumber = phoneNumber;

}

}

11.

**Program:**

**public** **class** BasicMath {

**public** **static** **int** add(**int** a, **int** b) {

**return** a + b;

}

}