JavaClass22 Method Overriding

Method Overriding:

Allows us to have same name methods in parent and child classes.

Why we should Override a method:

Lets say we are inheritaning from a Parent class that has 10 methods. we like 8 of them. we want to reuse those

methods, but we don't like 2 of them we want to provide implementation in a different way. We can use method overriding.

to override those methods.

what is the syntax to override a method:

we simply redefine the method in the child class.

package class22;

public class Animal {

void eat(){

System.out.println("Animals usually eat plants");

}

void sleep(){

System.out.println("Animals sleep for 8 to 10 hours");

}

}

class Panda extends Animal{

void eat(){

System.out.println("Pandas eat Bamboos");

}

/\*

@ovverirde is an annotation that tells java to check for extra rules like if we are properly overriding

a method or not

\*/

@Override

void sleep() {

System.out.println("Pandas sleep for 14 hours");

}

}

class Cat extends Animal{

void eat(){

System.out.println("Cats like to eat Fish");

}

}

package class22;

public class AnimalTester {

public static void main(String[] args) {

Cat cat=new Cat();

cat.sleep();

cat.eat();

Animal animal=new Panda(); // Upcasting

// Panda panda=new Animal(); // down-casting

Animal animal1=new Cat();

Animal [] animals={new Cat(),new Panda()};

}

}

package class22;

public class Car {

void start(){

System.out.println("Use the key to start me");

}

void stop(){

System.out.println("Use brakes to stop me");

}

void park(){

System.out.println("Park me manually");

}

}

class BMW extends Car{

void start(){

System.out.println("Use Push start to start me");

}

void stop() {

super.stop();

System.out.println("you can also use auto-breaking to stop me");

}

}

class Toyota extends Car{

void start(){

System.out.println("push me to start");

}

}

class Tesla extends Car{

void start(){

System.out.println("Use the App to start me"

);

}

void stop(){

System.out.println("Use AI and AutoBreaking to stop me");

}

void park(){

System.out.println("I can auto park myself");

}

}

package class22;

public class CarTester {

public static void main(String[] args) {

BMW bmw=new BMW();

bmw.start();

bmw.stop();

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

Toyota toyota=new Toyota();

toyota.start();

toyota.stop();

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

Tesla tesla=new Tesla();

tesla.stop();

\*/

}

}

package class22;

public class CastingDemo {

public static void main(String[] args) {

Animal animal=new Cat();

// Cat cat=(Cat)new Animal();

}

}

package class22;

public class CreditCardTester {

public static void main(String[] args) {

Visa visa=new Visa(100,5);

visa.calculateInterest();

}

// Break till 11:55

}

package class22;

public class FinalDemo {

public static void main(String[] args) {

Phone phone=new Phone();

// phone.RAM=50;

}

}

class Phone{

final int RAM=12;

void takePics(){

final int number=10;

}

}

final class Iphone extends Phone{

void takePics(){

}

}

package class22;

public class Parent {

static void print(){

System.out.println("Hello World");

}

}

class Child extends Parent{

static void print(){

System.out.println("Hello There");

}

}

class Test

{

public static void main(String[] args) {

Child.print();

}

}

package class22;

public class Task1 {

/\*

Create a class 'Degree' having a method 'getPrerequisite' that prints ""To get a degree you need high school diploma"".

Class 'Degree' has 2 subclasses namely 'Bachelors' and Masters’. In Masters class override method 'getPrerequisite'.

Call the method by creating an object of each of the three classes

\*/

public static void main(String[] args) {

Degree degree=new Degree();

degree.getPrerequisite();

Bachelors bachelors=new Bachelors();

bachelors.getPrerequisite();

Masters masters=new Masters();

masters.getPrerequisite();

}

}

class Degree{

void getPrerequisite(){

System.out.println("To get a degree you need high school diploma");

}

}

class Bachelors extends Degree{

}

class Masters extends Degree{

@Override

void getPrerequisite(){

System.out.println("To get a Master's degree you need Bachelors");

}

}

package class22;

public class Task2 {

/\*

Create a class CreditCard and define variable balance and interest.

Create an instance method that will calculate interest based on the given balance.

Create 2 subclasses: Visa and AX. In AX class override method calculate interest.

Call the method by creating an object of each of the three classes.

\*/

}

class CreditCard{

double balance;

double interest;

CreditCard(double balance,double interest){

this.balance=balance;

this.interest=interest;

}

void calculateInterest(){

double interestAmount=interest\*balance/100;

System.out.println("Interest Amount "+interestAmount);

}

}

class Visa extends CreditCard{

Visa(double balance,double interest){

super(balance, interest);

}

@Override

void calculateInterest() {

double interestAmount=interest\*balance/100;

System.out.println();

System.out.println("interest Amount "+(interestAmount+5));

}

}

class AX extends CreditCard{

AX(double balance,double interest){

super(balance, interest);

}

@Override

void calculateInterest() {

double interestAmount=interest\*balance/100;

System.out.println();

System.out.println("Interest Amount "+(interestAmount+10));

}

}

package class22;

public class WebDriver {

public void startBrowser(){

System.out.println("Starting the Browser");

}

public void test(){

System.out.println("Perform the testing");

}

public void closeBrowser(){

System.out.println("Closing the browser");

}

}

class Chrome extends WebDriver{

public void startBrowser(){

System.out.println("Starting the Google Chrome");

}

public void test(){

System.out.println("Perform the testing on Google Chrome");

}

public void closeBrowser(){

System.out.println("Closing the Google Chrome");

}

}

class FireFox extends WebDriver{

public void startBrowser(){

System.out.println("Starting the FireFox");

}

public void test(){

System.out.println("Perform the testing on FireFox");

}

public void closeBrowser(){

System.out.println("Closing the FireFox");

}

}

class Safari extends WebDriver{

public void startBrowser(){

System.out.println("Starting the Safari");

}

public void test(){

System.out.println("Perform the testing on Safari");

}

public void closeBrowser(){

System.out.println("Closing the Safari");

}

}

package class22;

public class WebDriverTester {

public static void main(String[] args) {

/\* Safari safari=new Safari();

safari.startBrowser();

safari.test();

safari.closeBrowser();

Chrome chrome=new Chrome();

chrome.startBrowser();

chrome.test();

chrome.closeBrowser();

FireFox fireFox=new FireFox();

fireFox.startBrowser();

fireFox.test();

fireFox.closeBrowser();\*/

WebDriver[] browsers={new Chrome(),new Safari(),new FireFox()};

for (WebDriver browser:browsers

) {

browser.startBrowser();

browser.test();

browser.closeBrowser();

}

for (int i = 0; i < browsers.length; i++) {

browsers[i].startBrowser();

browsers[i].test();

browsers[i].closeBrowser();

}

// Break till 1:50

/\* Chrome chrome=new Chrome();

FireFox fireFox=new FireFox();

Safari safari=new Safari();

WebDriver[] browsers={chrome,fireFox,safari};

\*/

/\*

Creating objects of all Child classes and storing them in an array of

type Webdriver we can do this bcause upcasting allows us to do this

\*/

}

}