**LAB 2. LỚP-ĐỐI TƯỢNG-KẾ THỪA**

1. **MỤC TIÊU**

Cung cấp cho sinh viên các kỹ thuật về lập trình hướng đối tượng trong JAVA:

- Xây dựng được một lớp hoàn chỉnh.

- Biết cách đặc tả truy xuất các thành phần bên trong lớp.

- Các tính chất của lập trình hướng đối tượng (kế thừa, nạp chồng, ghi đè phương thức)

- Hiểu và áp dụng được cách viết code kế thừa và đa hình trên Java.

- Hiểu và áp dụng được cách sử dụng mô hình lớp với mô tả kế thừa.

1. **NỘI DUNG**

- Khai báo sử dụng lớp, đối tượng.

- Tạo mảng các đối tượng, giải quyết các bài toán quản lý.

- Viết các chương trình sử dụng tính kế thừa.

1. **YÊU CẦU PHẦN CỨNG, PHẦN MỀM**

- Máy tính cài HĐH win**do**ws, RAM tối thiểu256MB.

- Phần mềm NETBEAN IDE 8.0, JDK 1.8

**D. KẾT QUẢ SAU KHI HOÀN THÀNH**

- Xây dựng được một lớp hoàn chỉnh.

- Biết cách sử dụng các đặc tả để truy xuất các thành phần của lớp.

- Xây dựng được ứng dụng với nhiều lớp được tổ chức theo sự phân cấp kế thừa

- Sử dụng lại những gì đã có ở một lớp khác

- Ghi đè hiệu chỉnh lại nội dung của một phương thức ở lớp con.

**E. HƯỚNG DẪN CHI TIẾT**

**1. Lớp**

* Tên lớp: Viết hoa
* Danh sách các thuộc tính (khai báo là **private**)
* Phương thức khởi tạo: Để khởi tạo đối tượng (hàm tạo 0, 1, 2… tham số)
* Danh sách các phương thức được cài đặt thêm

**Mô tả lớp:**

|  |
| --- |
| package pack.name;  Modifier class ClassName{  class’s body  } |

**Trong đó:**

pack.name : tên gói

Modifier : chỉ định truy cập (chỉ có thể là public hoặc không chỉ định)

ClassName: tên lớp, theo quy tắc định danh của Java

**Khai báo mảng đối tượng:**

|  |
| --- |
| ClassName [ ] ob= new ClassName [size]. |

**Bài 1.** Cài đặt lớp StudentMark gồm các thông tin:

* Rollnumber
* Họ tên
* Lớp
* Môn
* Điểm.

Cài đặt đầy đủ: 2 constructor, các phương thức set/get.

Cài đặt hàm nhập, hiển thị.

Khai báo luôn hàm main trong lớp này:

Khai báo 2 đối tượng của lớp, khởi tạo 2 đối tượng bằng construcor không có tham số. Gọi hàm nhập để nhập vào các thông tin và hàm hiển thị để hiển thị các thông tin.

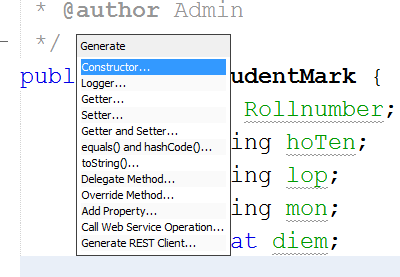
Hiển thị thông tin của người có điểm cao nhất.

**Hướng dẫn:**

Tạo class gồm các thuộc tính như đề bài

|  |
| --- |
| public class StudentMark {  private int Rollnumber;  private String hoTen;  private String lop;  private String mon;  private float diem;  } |

**Chèn tự động các hàm tạo, hàm set, get các thuộc tính vào class**



**Sau đó viết hàm nhap() và hàm hienthi() như sau:**

|  |
| --- |
| public void nhap() {  Scanner sc = new Scanner(System.in);  System.out.println("Nhap vao Rollnumber");  Rollnumber = Integer.parseInt(sc.nextLine());  System.out.println("Nhap vao ho ten :");  hoTen = sc.nextLine();  System.out.println("Nhap vao ten lop :");  lop = sc.nextLine();  System.out.println("Nhap vao ten mon hoc :");  mon = sc.nextLine();  boolean check = false;  do {  try {  System.out.println("Nhap vao diem :");  diem = Float.parseFloat(sc.nextLine());  check = true;  if (diem < 0 || diem > 10) {  check = false;  }  } catch (java.lang.NumberFormatException e) {  System.out.println("hay nhap dung kieu du lieu" + e.toString());  check = false;  }  } while (check == false);  }  public void hienthi() {  System.out.println("rollnumber :" + Rollnumber);  System.out.println("ho ten :" + hoTen);  System.out.println("ten lop :" + lop);  System.out.println("ten mon hoc :" + mon);  System.out.println("Nhap diem :" + diem);  }  } |

Tạo class **TestStudenMark.java**, khai báo hai đối tượng thuộc lớp SudentMark và sử dụng

|  |
| --- |
| public class TestStudenMark {  public static void main(String[] args) {  StudentMark st = new StudentMark();  st.nhap();  StudentMark stq = new StudentMark(1, "Nguyen Quyen", "abc", "tin", 9);  System.out.println("thong tin sinh vien 1;");  st.hienthi();  System.out.println("thong tin sinh vien 2 :");  stq.hienthi();  //tim kiem nguoi co diem cao nhat  StudentMark stTest = (st.getDiem() > stq.getDiem()) ? st : stq;  System.out.println("thong tin nguoi co diem lon nhat la :" );  stTest.hienthi();  }  } |

**Bài 2.**

Tạo **class** Product gồm các thuộc tính:

* Tên hàng hóa
* Nhà sản xuất
* Giá bán

+ Tạo 2 constructor cho lớp này.

+ Cài đặt hàm nhập và hiển thị.

Tạo class ProductMenu, khai báo hàm main và tạo menu sau:

1. Nhập thông tin cho n sản phẩm
2. Hiển thị thông tin vừa nhập
3. Sắp xếp thông tin giảm dần theo giá và hiển thị
4. Thoát

**Hướng dẫn:**

**Bước 1**: Tạo lớp Product gồm các thuộc tính.

**private**  String tenHangHoa;

**private** String nhaSanXuat;

**private** float giaBan;

Chèn tự động hàm tạo, hàm set, get.

Viết hàm nhập và hàm xuất thông tin như bài 1.

Bổ sung hàm sắp xếp:

|  |
| --- |
| public void sort(Product[] b){  for (int i = 0; i < b.length - 1; i++) {  for (int j = i + 1; j < b.length; j++) {  if (b[i].giaBan > b[j].giaBan) {  Product tem = b[i];  b[i] = b[j];  b[j] = tem;  }  }  }  } |

**Bước 2**: Tạo lớp ProductMenu có nội dung như sau.

|  |
| --- |
| public class ProductMenu {  static void menu() {  / /Viết nội dung lựa chọn chương trình tương ứng với đề bài  }  public static void main(String[] args) {  int n = 0;  Product a = new Product();  Product[] product = null;  do {  menu();  System.out.println("Nhap vao lua chon cua ban :");  Scanner sc = new Scanner(System.in);  n = Integer.parseInt(sc.nextLine());  switch (n) {  case 1: {  int m;  System.out.println("Nhap vao n :");  m = Integer.parseInt(sc.nextLine());  product = new Product[m];  for (int i = 0; i < m; i++) {  product[i] = new Product();  product[i].nhap();  }  break;  }  case 2: {  if (product == null) {  System.out.println("ban chua nhap du lieu");  } else {  System.out.println("du lieu ban vua nhap la :");  for (int i = 0; i < product.length; i++) {  System.out.println("thong tin hang hoa thu " + (i + 1));  product[i].hienthi();  }  }  break;  }  case 3: {  if (product == null) {  System.out.println("ban chua nhap du lieu");  } else {  a.sort(product);  System.out.println("du lieu sau khi sap xep la :");  for (int i = 0; i < product.length; i++) {  System.out.println("thong tin hang hoa thu " + (i + 1));  product[i].hienthi();  }  }  break;  }  case 4:  break;  default: {  System.out.println("khong co lua chon cua ban ");  break;  }  }  } while (n != 4);  }  } |

**Bài 3.**

Cài đặt lớp Product gồm các thuộc tính (phải khai báo là **private**)

* String maHH;
* String tenHH;
* float soLuong;
* float gia1SP;

Cài đặt 2 construcors, các hàm get/set.

Cài đặt hàm input(), display().

Khai báo hàm main và thực hiện như sau:

* Khai báo mảng có n phần tử kiểu Product.
* Gọi nhập thông tin cho các phần tử của mảng.
* Tìm ra sản phẩm nào có giá bán cao nhất.
* Sắp xếp theo thứ tự giảm dần của giá
* Tìm trong danh sách hàng hóa có mặt hàng “Sữa” hay không?

**Hướng dẫn:**

**Bước 1**: Tương tự bài 2, xây dựng lớp ProductBai3 gồm các