BÁO CÁO THỰC HÀNH LAB 03
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

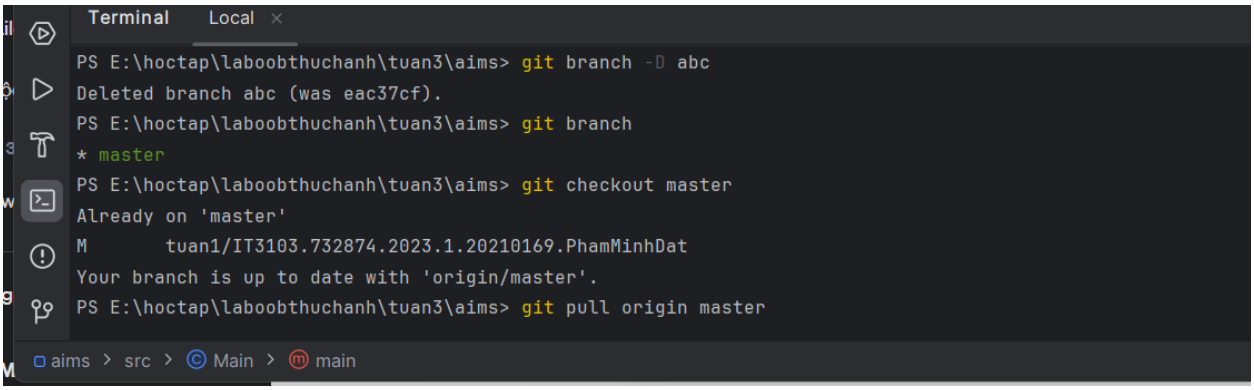
Phạm Minh Đạt -20210169

Mục lục

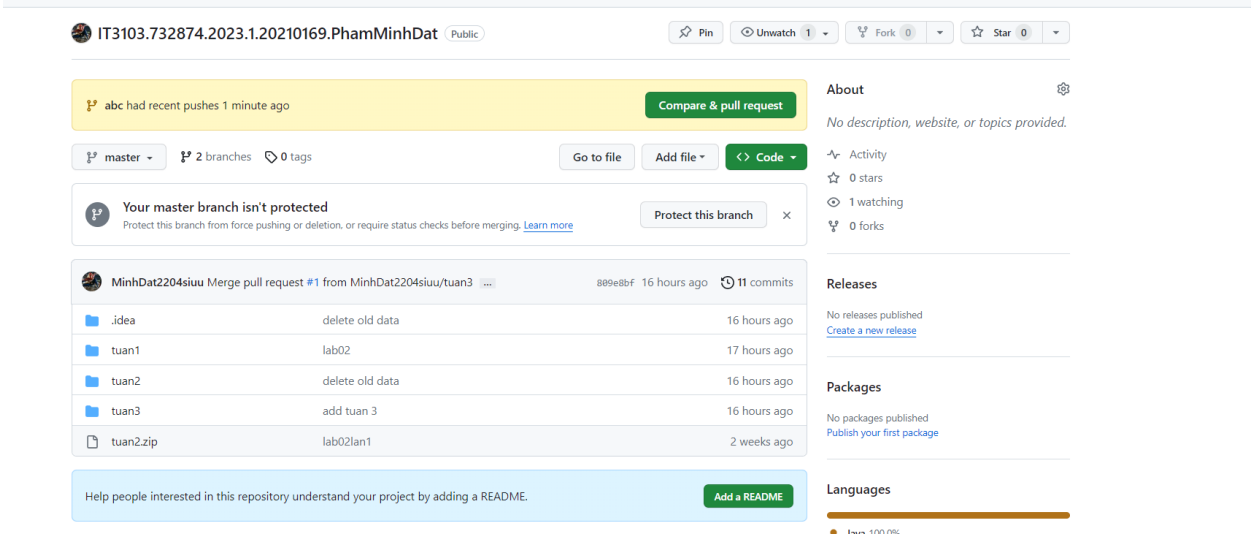
Mục Lục

Mục Lục ảnh

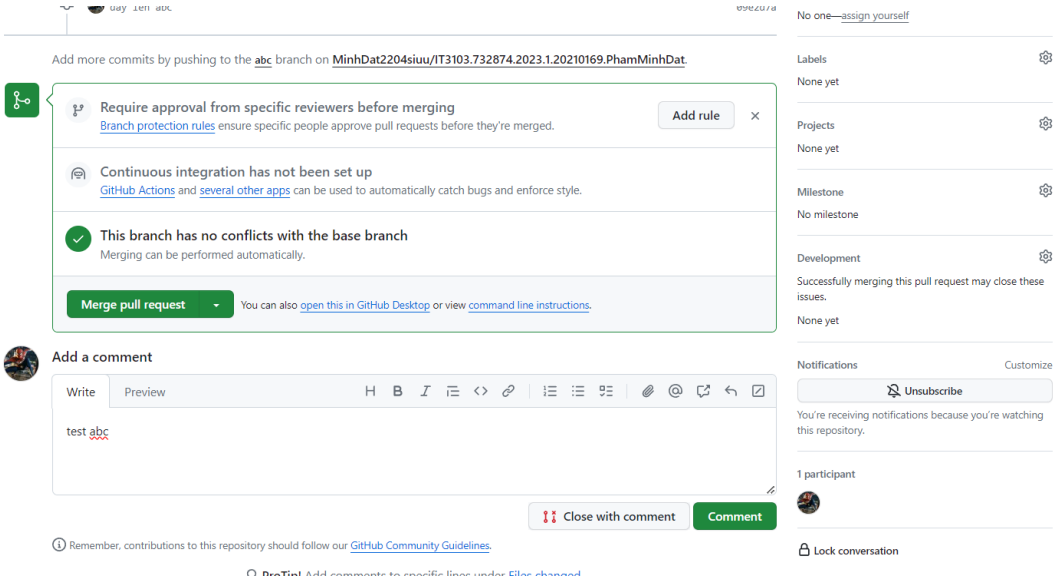
1.Branch your repository



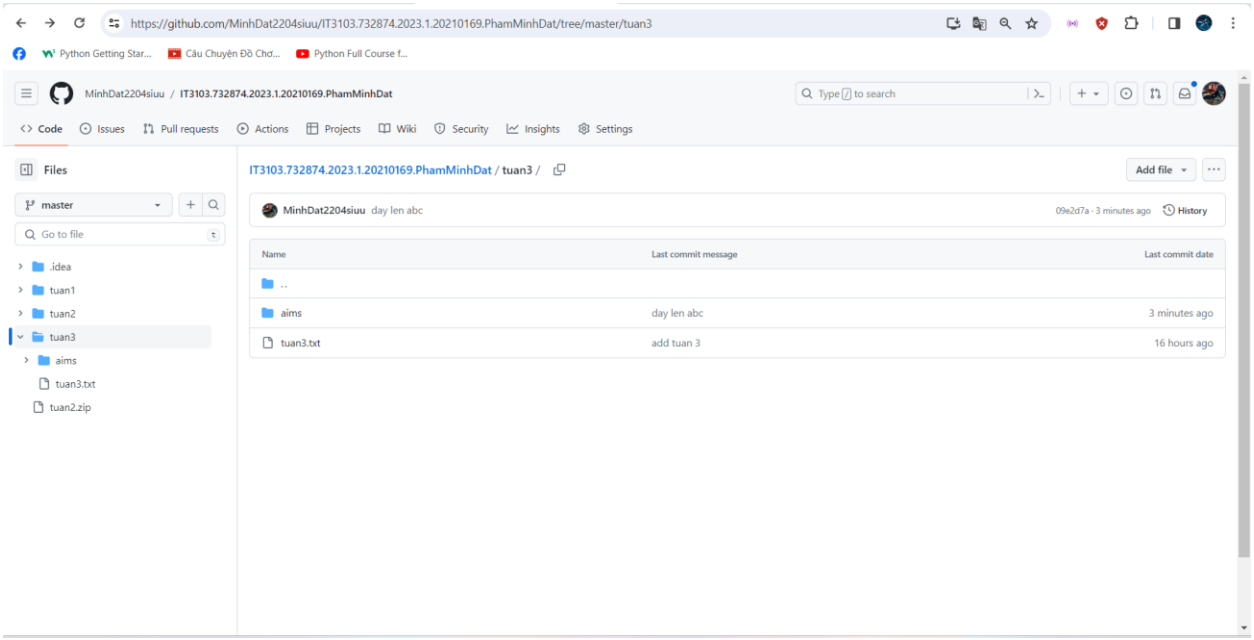
Pic1. Tạo branch mới abc và checkout vào để làm việc



Pic2: push lên branch abc



Pic 3: merge branch abc vào master



Pic4: đã merge thành công

```
PS E:\hoctap\laboobthuchanh\tuan3\aims> git checkout master
Switched to branch 'master'
M       tuan1/IT3103.732874.2023.1.20210169.PhamMinhDat
Your branch is up to date with 'origin/master'.
PS E:\hoctap\laboobthuchanh\tuan3\aims> git merge abc
Updating 809e8bf..09e2d7a
Fast-forward
```

Pic5: ta merge lại nhánh abc vào nhánh master trên máy tính .

2.Working with method overloading

```
46
    no usages   new *   3 related problems
47 @    public boolean addDigitalVideoDisc(DigitalVideoDisc[] dvdList){
48      int n = dvdList.length;
49      if (qtyOrdered + n <= MAX_NUMBERS_ORDERED){
50      for (int i = 0; i < MAX_NUMBERS_ORDERED; i++){
51      for (int j = 0; j < n; j++){
52      if (itemsOrdered[i] == null){
53          itemsOrdered[i] = dvdList[j];
54      }
55      }
56      } return true;
57      } else {return false;}
58      }
```

Pic6: Method addDigitalVideoDisc

```
                                                          31  17  1
    no usages   minh dat *   3 related problems
public boolean addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {
    boolean place1 = false;
    if (qtyOrdered + 2 <= MAX_NUMBERS_ORDERED) {
        for (int i = 0; i < MAX_NUMBERS_ORDERED; i++){
            if ((!place1) && (itemsOrdered[i] != null)) {
                itemsOrdered[i] = dvd1;
                place1 = true;
            } else if (place1 && (itemsOrdered[i] == null)){
                itemsOrdered[i] = dvd2;
                break;
            }
        } return true;
    } else {
        return false;
    }
}

    2 usages   new *
```

Pic 7: Method addDigitalVideoDisc(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2)

3.Passing parameter

Aims.java | TestPassingParameter.java

```java
package hust.soict.dsai.test.disc;
import hust.soict.dsai.aims.disc.DigitalVideoDisc;
public class TestPassingParameter {
    public static void main(String[] args){
        DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
        DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderealla");

        DigitalVideoDisc[] swapped = swap(jungleDVD, cinderellaDVD);
        jungleDVD = swapped[0];
        cinderellaDVD = swapped[1];

        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
        System.out.println("cinderella dvd title: " + cinderellaDVD.getTitle());

        changeTitle(jungleDVD, cinderellaDVD.getTitle());
        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
    }

    // public static void swap(Object o1, Object o2) {
    //     Object tmp = o1;
```

Pic 8: Passing parameter code

```java
1 usage
public static void changeTitle(DigitalVideoDisc dvd, String title) {
    String oldTitle = dvd.getTitle();
    dvd.setTitle(title);
    dvd = new DigitalVideoDisc(oldTitle);
}
```

Pic 9: Passing parameter code

```
jungle dvd title: Cinderella
cinderella dvd title: Jungle
jungle dvd title: Cinderella

Process finished with exit code 0
```

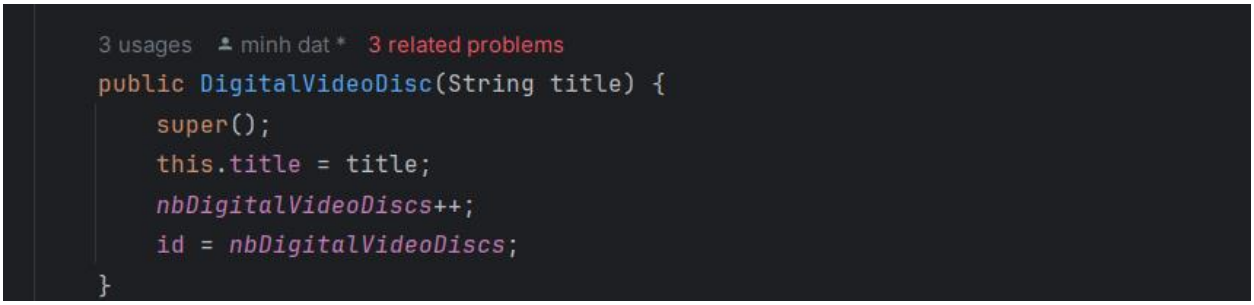
Pic 10: Passing parameter code

4. Use debug run

Chạy debug giúp ta hiểu rõ hơn sự thực thi của chương trình, đồng thời ta còn có thể thay đổi giá trị của object một cách thủ công.

5. Classifier Member and Instance Member

Pic 11: Classifier Member and Instance Member Code



Pic 12: Classifier Member and Instance Member Code

```java
Cart anOrder = new Cart();

// Tạo đối tượng DVD mới và thêm chúng vào giỏ hàng
DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King",
        "Animation","Roger Allers", 87, 19.95f);
anOrder.addDigitalVideoDisc(dvd1);
DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars",
        "Science Fiction","George Lucas", 87, 24.95f);
anOrder.addDigitalVideoDisc(dvd2);
DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladdin",
        "Animation", 18.99f);
anOrder.addDigitalVideoDisc(dvd3);

// In ra tổng giá trị của các sản phẩm trong giỏ hàng
System.out.print("Total cost it: ");
System.out.println(anOrder.totalCost());

// Kiểm tra phương thức remove
anOrder.removeDigitalVideoDisc(dvd3);
System.out.print("Total cost is: ");
```

Pic 13: Classifier Member and Instance Member Code

```
The disc has been added
The disc has been added
The disc has been added
Number of dvd is:
3
Id of dvd3 is:
3

Process finished with exit code 0
```

Pic 14: Classifier Member and Instance Member result

6. Open the Cart class

```
public void print() {
    System.out.println("*".repeat( count: 15) + "CART" + "*".repeat( count: 15));

    int[] ids = new int[MAX_NUMBERS_ORDERED];
    for (int i = 0; i < MAX_NUMBERS_ORDERED; i++){
        DigitalVideoDisc curr_disc = itemsOrdered[i];
        if (curr_disc != null) {
            ids[i] = curr_disc.id;
        }
    }
    Arrays.sort(ids);

    for (int i = 0; i < MAX_NUMBERS_ORDERED; i++) {
        int id = ids[i];
        if (id > 0) {
            System.out.println(searchID(id));
        }
    }
    System.out.println("*".repeat( count: 34));
}
```

Pic 14: Method in class Cart to print the list

```
no usages  minh dat
public void searchTitle(String title) {
    for (int i = 0; i < MAX_NUMBERS_ORDERED; i++){
        DigitalVideoDisc curr_disc = itemsOrdered[i];
        if (curr_disc != null) {
            if (curr_disc.getTitle() == title){
                System.out.println(toString(curr_disc));
            }
        }
    }
}
```

Pic15: Method in class Cart to search Title

```
public String searchID(int id){
    boolean found = false;
    for (int i = 0; i < MAX_NUMBERS_ORDERED; i++){
        DigitalVideoDisc curr_disc = itemsOrdered[i];
        if (curr_disc != null) {
            if (curr_disc.id == id) {
                found = true;
                return toString(curr_disc);

            }
        }
    } return "No match found";
}
```

Pic16: Method in class Cart to search ID

7. Implement the Store class

```
3 usages  minh dat
public class Store {
    5 usages
    private int nbDVD = 1000;
    5 usages
    private int qtyDVD = 0;
    5 usages
    private DigitalVideoDisc itemsInStore[] = new DigitalVideoDisc[nbDVD];

    3 usages  minh dat
    public boolean addDVD(DigitalVideoDisc disc){
        for (int i = 0; i < nbDVD; i++){
            if ((itemsInStore[i] == null) && (qtyDVD < nbDVD)) {
                itemsInStore[i] = disc;
                qtyDVD++;
                return true;
            }
        } return false;
```

Pic17: Code class Store

```
1 usage  minh dat
public boolean removeDVD(DigitalVideoDisc disc) {
    for (int i = 0; i < nbDVD; i++){
        if ((itemsInStore[i] != null) && (qtyDVD > 0)) {
            DigitalVideoDisc curr_disc = itemsInStore[i];
            if (curr_disc.equals(disc)) {
                itemsInStore[i] = null;
                qtyDVD--;
                return true;
            }
        }
    } return false;
}

1 usage  minh dat
public Store(int nbDVD){
    super();
    this.nbDVD = nbDVD;
}
```

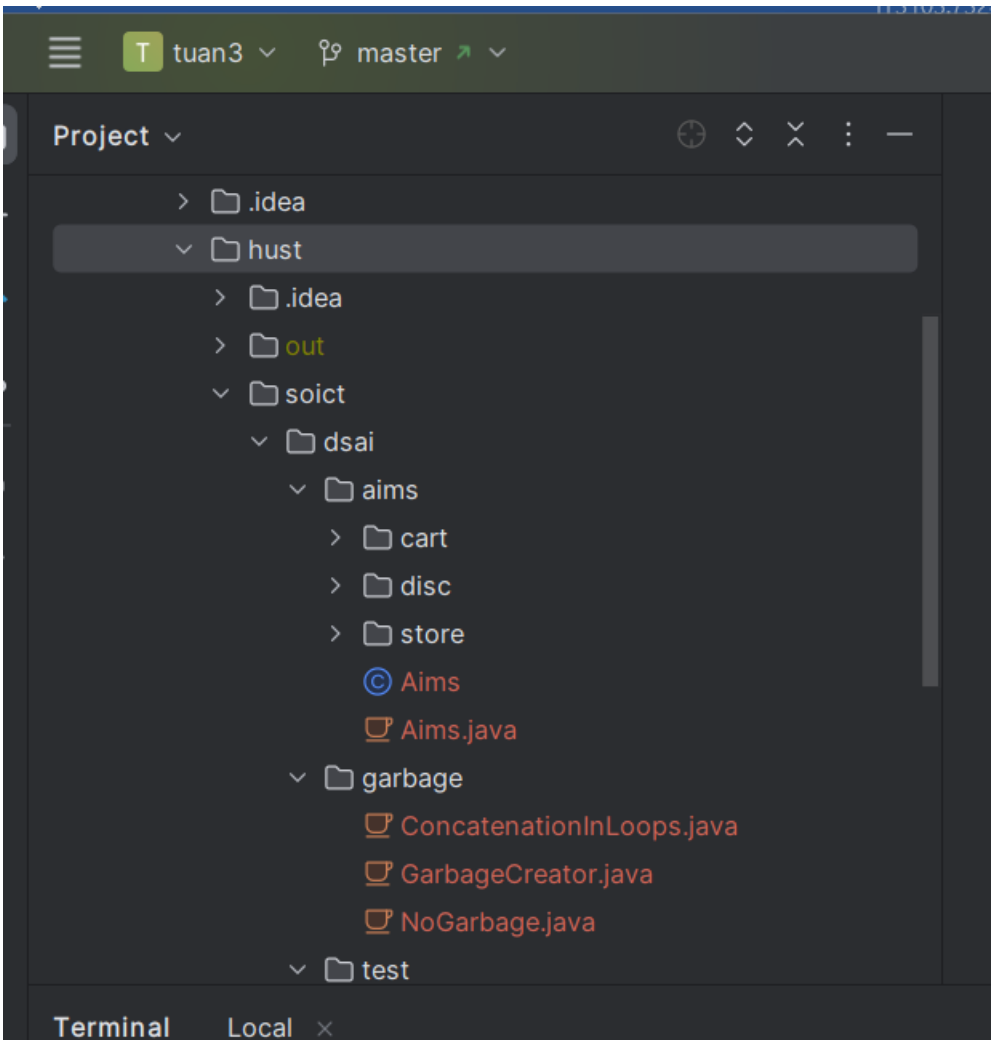
Pic 18: Code class Store

```
        this.nbDVD = nbDVD;
    }

    // for testing method
    2 usages  minh dat
    public int getQty(){
        return qtyDVD;
    }
}
```

Pic 19: Code class Store

8. Re-organize your projects



Pic 20 sort folder code

9. String, StringBuilder and StringBuffer

```java
public class ConcatenationInLoops {

    public static void main(String[] args){
        Random r = new Random(123);
        long start = System.currentTimeMillis();
        String s = "";

        for (int i = 0; i < 65536; i++){
            s += r.nextInt(2);
        } System.out.println(System.currentTimeMillis() - start);

        r = new Random(123);
        start = System.currentTimeMillis();
        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < 65536; i++){
            sb.append(r.nextInt(2));
        }
        s = sb.toString();
        System.out.println(System.currentTimeMillis() - start);
    }
}
```

Pic 21 Code ConcatenationInLoops

```java
public class GarbageCreator {

    public static void main(String[] args) {
        String filename = "OtherProjects/test.txt";
        byte[] inputBytes = { 0 };
        long startTime, endTime;

        try {
            inputBytes = Files.readAllBytes(Paths.get(filename));
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        startTime = System.currentTimeMillis();
        String outputString = "";
        for (byte b : inputBytes) {
            outputString += (char)b;
        }
        System.out.println(System.currentTimeMillis() - startTime); // 17
    }
}
```

Pic 22 Code Garbagecreator

Pic23 NoGarbage code

Question: Is JAVA a Pass by Value or a Pass by Reference programming language?

=> In Java, when you pass a variable to a method, you are passing the value of the variable, not the actual variable itself. This is known as "pass-by-value."

In the context of method parameters, passing by value means that a copy of the value stored in the variable is passed to the method. Any changes made to the parameter inside the method do not affect the original variable outside the method.