Lab09

Name: Tet Davann

ID: IDTB080023

```
import java.util.ArrayList;
import java.util.Scanner;

abstract class SubTask{
    int duration;
    String name;
}
class TaskProcess implements Runnable{
    SubTask task;
    public TaskProcess(SubTask task){
        this.task = task;
    }
    private void doProcess(){
        int k=1000;
            do{
                if(k/task.duration==1){
                    System.out.println(task.duration/1000+"s ::: "+task.name+" -
100.00% - completed :::");
                }else{
                    System.out.printf("%d%s%s %.2f%s",k/1000,"s :::
",task.name,(float)k*100/ task.duration,"% - processing :::");
                    System.out.println();
                }

                try {
                    Thread.sleep(1000);
                    k+=1000;
                } catch (InterruptedException e) {
                    throw new RuntimeException(e);
                }
            }while(k<=task.duration);

    }
    @Override
    public void run() {
        doProcess();
    }


}
public class Lab09_1 extends Thread{
    private static Scanner sc = new Scanner(System.in);
    private static ArrayList<SubTask> listProcess = new ArrayList<>();
    private static void initProcess(){
        int[] arrayDuration = {1000,6000,8000};
        String[] arrayName = {"Format hard disk","Download video","Develop an app"};
        SubTask subTask;
        for(int i=0;i<arrayName.length;i++){
            subTask = new SubTask(){};
            subTask.name = arrayName[i];
```

```
            subTask.duration = arrayDuration[i];
            listProcess.add(subTask);
        }
    }
    private static void menu(){
        int i=1;
        System.out.println("================================================");
        System.out.println("| No\t| Name\t\t\t\t\t| Duration\t\t|");
        System.out.println("================================================");
        for(SubTask task:listProcess){
            System.out.println("| "+i+"\t\t"+"| "+task.name+"\t\t"+"|
"+task.duration+"ms\t\t|");
            i++;
        }
        System.out.println("================================================\n");
        System.out.print("1. Add new task\n" +
                "2. Delete a task\n" +
                "3. Proceed tasks\n" +
                "Choose an opt:");
        int opt = sc.nextInt();
        switch (opt){
            case 1->addTask();
            case 2-> deleteTask();
            case 3->proccedTasks();
        }
        menu();
    }
    private static void addTask(){
        System.out.println("Add new task");
        System.out.print("Name: ");
        sc.nextLine();
        String names = sc.nextLine();
        System.out.print("Duration(ms): ");
        int durations = sc.nextInt();
        SubTask subTask = new SubTask(){};
        subTask.name = names;
        subTask.duration = durations;
        listProcess.add(subTask);
    }
    private static void deleteTask(){
        System.out.println("Delete a task");
        System.out.print("No: ");
        int no = sc.nextInt();
        if(no>0&&no<=listProcess.size()){
            listProcess.remove(no-1);
        }else{
            System.out.println("No is invalid!please try again");
            deleteTask();
        }
    }
    private static void proccedTasks(){
        ArrayList<TaskProcess> listTaskProcess = new ArrayList<>();
        System.out.println("\tProcced tasks");
        System.out.println("Select tasks to perform parallel process:");
        int no;
        String ch;
        do{
            ch="";
```

```
            System.out.print("Input task No: ");
            no = sc.nextInt();
            if(no>0&&no<=listProcess.size()){
                listTaskProcess.add(new TaskProcess(listProcess.get(no-1)));
                System.out.print("Do you want to add more task (y/n)?: ");
                ch = sc.next();
            }
        }while (ch.toLowerCase().charAt(0)=='y');
        Thread[] thread = new Thread[listTaskProcess.size()];
        int i = 0;
        for(TaskProcess testTask : listTaskProcess){
            thread[i] = new Thread(testTask,"Thread-1");
            thread[i].start();
            i++;
        }
        for(Thread threads:thread){
            try {
                threads.join();
            } catch (InterruptedException e) {
                throw new RuntimeException(e);
            }
        }
    }
    public static void main(String[] args){
        initProcess();
        menu();
    }
}
```

```
1. Add new task
2. Delete a task
3. Proceed tasks
Choose an opt:1
Add new task
Name: Install VsCode
Duration(ms): 10000
================================================
| No    | Name                  | Duration     |
================================================
| 1     | Format hard disk      | 1000ms       |
| 2     | Download video        | 6000ms       |
| 3     | Develop an app        | 8000ms       |
| 4     | Install VsCode        | 10000ms      |
================================================
```

```
1. Add new task
2. Delete a task
3. Proceed tasks
Choose an opt:2
Delete a task
No: 4
================================================
| No    | Name                  | Duration     |
================================================
| 1     | Format hard disk      | 1000ms       |
| 2     | Download video        | 6000ms       |
| 3     | Develop an app        | 8000ms       |
================================================
```

```
Select tasks to perform parallel process:
Input task No: 1
Do you want to add more task (y/n)?: y
Input task No: 2
Do you want to add more task (y/n)?: y
Input task No: 3
Do you want to add more task (y/n)?: n
1s ::: Download video 16.67% - processing :::
1s ::: Develop an app 12.50% - processing :::
1s ::: Format hard disk - 100.00% - completed :::
2s ::: Download video 33.33% - processing :::
2s ::: Develop an app 25.00% - processing :::
3s ::: Develop an app 37.50% - processing :::
3s ::: Download video 50.00% - processing :::
4s ::: Develop an app 50.00% - processing :::
4s ::: Download video 66.67% - processing :::
5s ::: Download video 83.33% - processing :::
5s ::: Develop an app 62.50% - processing :::
6s ::: Develop an app 75.00% - processing :::
6s ::: Download video - 100.00% - completed :::
7s ::: Develop an app 87.50% - processing :::
8s ::: Develop an app - 100.00% - completed :::
```