RT- Java Compte rendu

Nechi Aziz & Ghaith Nabli

Exercice 2:

Code de la classe Compte :

```java
package exercice2;

public class Compte {

   int solde ;
   public Compte(){
      this.solde = 100;
   }

   public Compte(int solde){
      this.solde = solde;
   }

   public boolean retirer(int montant){
      if (this.solde > montant) {
         this.solde -= montant;
         return true;
      }
      else
         return false;
   }
}
```

Code de la classe JobSylvieEtBruno :

```java
package exercice2;

public class JobSylvieEtBruno implements Runnable {
   Compte compte ;
   String nom ;
   int montant;
   public JobSylvieEtBruno(String nom, Compte compte, int montantDebit){
      this.nom = nom;
      this.compte = compte;
      this.montant = montantDebit;
   }
   @Override
   public void run() {
      int i = 10;
      while(i>0) {
         i--;
         if (this.compte.retirer(this.montant)) {
            System.out.println(this.nom + " a débiter " + this.montant + "$");
         } else System.out.println(this.nom + " ne peut pas débiter: Solde insuffisant.");
         try {
            Thread.sleep(500);
         } catch (InterruptedException e) {
            e.printStackTrace();
         }
      }
   }
}
```

Les objets qui implémentent cette classes seront unique par leur nom, le compte qu'ils utilisent et le montant à débiter à chaque transaction (chaque 500ms)

Vu que Sylvie et Bruno ont un compte en commun, le programme suivant illustre le principe de ressource partagée.

Programme moteur :

```java
package exercice2;

public class Exercice2Main {
   public  static void main(String[]args) {
      Compte compte = new Compte();   // Par défaut, contient 100$.
      JobSylvieEtBruno sylvie = new JobSylvieEtBruno("Sylvie", compte, 5);
      JobSylvieEtBruno bruno = new JobSylvieEtBruno("Bruno", compte, 7);
      new Thread(sylvie).start();
      new Thread(bruno).start();
   }
}
```

Résultat de l'exécution :

```
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
```

```
Sylvie a débiter 5$
Bruno a débiter 7$
Sylvie a débiter 5$
Bruno ne peut pas débiter: Solde insuffisant.
Sylvie ne peut pas débiter: Solde insuffisant.
Bruno ne peut pas débiter: Solde insuffisant.
```

Le compte étant initialement crédité de 100$, chacun de Sylvie et Bruno aurait dû être capable d'effectuer tout ses retraits (10*7$ pour Bruno et 10*5$ pour Sylvie) .

Sauf qu'on a passer à tout les deux le même compte, chaque thread effectuent son débit parallèlement.