1. Soit un objet*television* et une méthode*allumer* définie pour cet objet, lequel des énoncés

suivants permet d’appeler la méthode allumer ?

* 1. **allumer.television()**
  2. **allumer.television**
  3. **television.allumer**
  4. **television.allumer()**

1. Réponse : \_\_\_\_\_\_\_\_\_\_\_ **television.allumer()** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Soit la classe **Incremente** suivante :

|  |  |
| --- | --- |
| **public class Incremente**  **{ private int increment; private int petitPlus;**    **public Incremente(int inc, int petit)**  **{**  **increment = inc;**  **petitPlus = petit;**  **}**    **public int additionne(int n)**  **{**  **int somme;**   |  | | --- | | ou tout simplement :  **return n + increment + petitPlus;** |   **somme = n + increment + petitPlus;**  **return somme;**  **}**  **}** |

Quel résultat va-t-on obtenir suite à l’exécution de l’application suivante :

**public class Test**

**{**

**public static void main(String args[])**

**{**

**Incremente unObjet = new Incremente(10, 1);**

**System.out.println("" + unObjet.additionne(5));**

**}**

**}**

Résultat : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 16 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Soit la classe suivante :

|  |
| --- |
| **public class Cours**  **{ private String sigle; // sigle du cours private int eval; // évaluation du cours**    **public Cours(String sigle, int eval) // constructeur**  **{**  **.......**  **}**    **public String appreciation() // méthode d’instance**  **{**  **.......**  **}**  **}** |

* 1. Complétez la déclaration de l’objet *cours2345* qui instancie la classe Cours avec "INFO 2345" comme sigle du cours et la valeur 3 comme évaluation :

**\_\_\_\_\_\_\_\_Cours\_ cours2345 = new Cours(“INFO 2345", 3)\_\_\_\_\_\_\_\_**

* 1. Complétez le code du constructeur*Cours* en tenant compte de ses paramètres :

**public Cours(String sigle, int eval)**

**{**

**this.sigle = sigle;**

**this.eval = eval;**

**}**

* 1. En supposant que, dans la méthode main, on ait l'instruction suivante :

**System.out.println("J'aime Java " + cours2345.appreciation());**

|  |  |
| --- | --- |
| **eval** | **message** |
| 1 | pas du tout |
| 2 | un peu |
| 3 | beaucoup |
| 4 | à la folie |

Donnez le code de la méthode *appreciation* qui retourne un message d’appréciation selon la valeur de l’attribut *eval* et en tenant compte du tableau suivant :

**public String appreciation() {**

String texte;

Switch (this.eval) {

case 1: texte = ‘’pas du tout’’; break;

case 2: texte = ‘’un peu’’; break;

case 3: texte = ‘’beaucoup’’; break;

case 4: texte = ‘’à la folie’’; break;

}

return texte;

**}**

1. On a les classes suivantes :

* 1. **public class Livre**
  2. **{**
  3. **private static int nbLivres = 0;**
  4. **private String titre;**
  5. **private int nbPages;**
  6. **private int annee;**

**7**

# 8 // constructeur d’initialisation

1. **public Livre(String t, int nb, int a)**
2. **{**
3. **titre = t;**
4. **nbPages = nb;**
5. **annee = a;**
6. **nbLivres++;**
7. **}**

**16**

1. **private int getNbPages()**
2. **{ return nbPages; }**

**19**

1. **public static String getTitre()**
2. **{ return titre; }**

**22**

1. **public int getAnnee()**
2. **{ return annee; }**

**25**

1. **public String affiche()**
2. **{**
3. **return titre + ", paru en " + annee + ", a " + nbPages + " pages";**
4. **}**

# 30 } // fin Livre

1. **public class TestLivre**
2. **{**
3. **public static void main(String args[])**
4. **{**
5. **Livre dictionnaire;**
6. **Livre roman = new Livre("Da Vinci code", 514, 2003);**
7. **Livre essai = new Livre();**

**46**

**48 System.out.println("Le roman a " + roman.getNbPages());**

**49**

**50 System.out.println("Le titre du roman est " + roman.getTitre());**

**51**

1. **System.out.println("Le dictionnaire est paru en " +**
2. **dictionnaire.getAnnee());**

**54**

**55 System.out.println(roman.affiche());**

**56**

**57 System.out.println(roman);**

**58**

1. **System.out.println("Le nombre de livres est " + Livre.nbLivres);**
2. **}**

# 61 } // fin TestLivre

Expliquez les problèmes obtenus lors de la compilation (ou de l'exécution) des lignes suivantes :

|  |  |
| --- | --- |
| **21** |  |
| **46** |  |
| **48** |  |
| **50** |  |
| **53** |  |
| **57** |  |
| **59** |  |

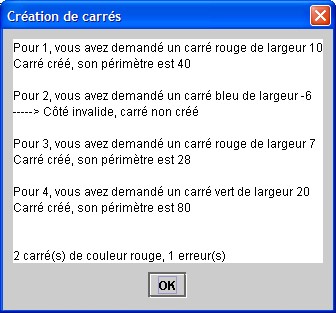
5. Soit la classe suivante :

|  |  |  |  |
| --- | --- | --- | --- |
| **{** | **public class Carre**  **private String couleur;**  **private int cote;** |  | **// longueur d’un côté du carré** |
|  | **private int perimetre; public Carre(String coul, int nb)** |  | **// constructeur** |
| **}** | **{**  **couleur = coul; cote = nb;**  **}**  **public void calculerPerimetre()**  **{**  **perimetre = cote \* 4;**  **}**  **public String getCouleur() { return couleur; }**  **public int getCote() { return cote; }**  **public int getPerimetre()**  **{ return perimetre; }** |  |  |

Complétez la classe **TestCarre** afin de produire l'affichage ci-dessous. On doit :

* saisir au clavier la couleur et le côté de 4 objets Carre
* afficher la couleur et le côté de chaque carré demandé
* rejeter une valeur négative pour le côté en affichant un message approprié (on ne crée pas l'objet dans ce cas)
* calculer et afficher le périmètre de chaque objet Carre créé
* comptabiliser le nombre de carrés de couleur rouge et l’afficher
* comptabiliser le nombre d’erreurs et l’afficher

**Résultats de l’exécution :**



**import javax.swing.\*; public class TestCarre**

**{**

**public static void main(String args[])**

**{**

**final int NB\_CARRES = 4;**

**JTextArea affichage = new JTextArea();**

**String laCouleur; // pour lire la couleur d'un carré int longCote; // pour lire la longueur du côté int nbRouge = 0; // compteurs int nbErreurs = 0;**

**Carre unCarre; // déclaration d'une référence à un objet Carre**

**for (int k = 1; k <= NB\_CARRES; k++)**

**{**

**laCouleur = JOptionPane.showInputDialog( "Entrez la couleur du carré " + k); longCote = Integer.parseInt(JOptionPane.showInputDialog( "Entrez la longueur du côté pour le carré " + k)); affichage.append("Pour " + k + ", vous avez demandé un carré " + laCouleur + " de largeur " + longCote + "\n");**

**// validation du côté**

**if (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)**

**{**

**// erreur**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**}**

**else**

**{**

**// création de l'objet unCarre**

**// calcul et affichage de son périmètre, etc.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**}**

**}**

**// affichage des résultats**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**JOptionPane.showMessageDialog(null, affichage,**

**"Création de carrés", JOptionPane.PLAIN\_MESSAGE);**

**System.exit(0);**

**}**

**}**

1. Créez une classe **Auto** avec :
   * + les attributs *marque*, *couleur*, *bonEtat*
     + le constructeur qui affecte la marque et la couleur
     + les méthodes *rouler* (met bonEtat à true), *horsCircuit* (met bonEtat à false) et *toString* (qui retourne une phrase avec la marque, la couleur et l’état du véhicule (défectueux ou fonctionnel)

Créez aussi une classe **TestAuto** qui crée une Volvo noire et qui fait l'appel de la méthode *rouler* suivi de l’affichage de l'objet (à l'aide de *toString*), puis de l’appel de *horsCircuit* et de l'affichage de l'objet à nouveau. Quels seront les résultats affichés ?

**Résultats de l’exécution :**

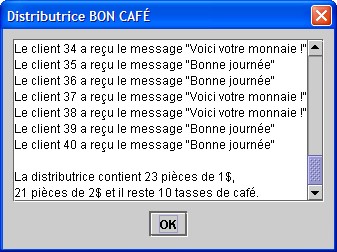
|  |
| --- |
|  |

1. Sachant qu’on voudra créer dans une application des machines distributrices qui ne vendent que du café coûtant 1$ et qu’on pourra acheter le café en payant seulement avec des pièces de 1$ ou de 2$ :
   1. Complétez la classe **Distributrice** avec :
      * ses 3 attributs
      * le constructeur sans paramètre et celui à 2 paramètres
      * les méthodes « *set* » et les méthodes « *get* »
      * la méthode *payerCafe* (qui doit retourner un des messages suivants : **Bonne journée** ou **Voici votre monnaie** ou **Désolé! je n’ai plus de 1$** ou **La pièce est refusée** ou **Désolé! il n’y a plus de café**)
      * la méthode *toString* qui retourne une chaîne donnant le contenu de la distributrice

* 1. Complétez le diagramme de la classe **Distributrice**

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | **TestDistributrice** | |  | | **Distributrice**    utilise |

* 1. Complétez l'application **TestDistributrice** qui va créer une distributrice contenant au départ 25 pièces de 1$ et 50 tasses de café. L'application doit ensuite simuler l’achat de café par 40 clients à l’aide d’une boucle **for** (le client paie avec un 1$ lorsque le compteur de la boucle est un multiple de 3 ou de 5, sinon il paie avec un 2$). Faites afficher les messages concernant les ventes de café et l’état de la distributrice à la fin de la journée dans une zone de texte à défilement sur 10 lignes.



**public class Distributrice**

**{**

***// constantes***

**public**

**static**

**final**

**int**

**UN\_DOLLAR = 1;**

**public**

**static**

**final**

**int**

**DEUX\_DOLLARS = 2;**

***// attributs d'instance***

***// constructeurs***

***// méthodes d'accès (get)***

***// méthodes de mutation (set)***

***// méthode toString***

***// méthode payerCaf***

***e***

**import javax.swing.\*;**

**public class TestDistributrice**

**{**

**public static void main(String[] args)**

**{**

**JTextArea affichage = new JTextArea(10, 0);**

**JScrollPane d**

**e**

**filement = new JScrollPane(affichage);**

***// cré***

***ation de la***

***D***

***istributrice***

***// boucle d'achat de café par 40 clients***

***// affichage des ventes et de l'état de la Distributrice***

**JOptionPane.showMessageDialog(null, d**

**e**

**filement,**

**"Distributrice BON CAFÉ", JOptionPane.P**

**LAIN\_MESSAGE);**

**System.exit(0);**

**}**

**}**

1. Supposons que la classe **FouFou** soit définie comme suit :

|  |  |  |
| --- | --- | --- |
| **{** | **public class FouFou**  **private int nb;**  **private static String nom;**  **public int getNb()** | **{ return nb; }** |
|  | **public static String getNom()** | **{ return nom; }** |
|  | **public void methode1()** | **{ ......... ; }** |
| **}** | **public static void methode2()** | **{ ......... ; }** |

et soit la création de l'objet suivant : **FouFou unFou = new FouFou();** Indiquez si les instructions suivantes sont correctes ou non :

* + - 1. **System.out.println("Le nombre est " + unFou.getNb()); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      2. **System.out.println("Le nom est " + unFou.getNom()); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      3. **unFou.methode1(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      4. **unFou.methode2(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      5. **System.out.println("Le nombre est " + FouFou.getNb()); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      6. **System.out.println("Le nom est " + FouFou.getNom()); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      7. **FouFou.methode1(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**
      8. **FouFou.methode2(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. Complétez la classe **Exemple** suivante, sachant que les objets **Exemple**, qui seront créés par la suite, doivent partager les attributs *somme* et *nbObjets*. L'attribut *nbObjets* permet de compter le nombre d'objets qui seront créés et l'attribut *somme* permet de faire la somme des différentes valeurs conservées dans l'attribut d'instance *valeur* des différents objets créés.

|  |
| --- |
| **public class Exemple**  **{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ nbObjets = 0;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ somme = 0;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ valeur;**  **public Exemple(int nb)**  **{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **}**  **public \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ setValeur(int nb)**  **{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **}**    **public \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ getValeur()**  **{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **}** |

**public \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ getNbObjets()**

**{**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**

**}**

**public \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ getSomme()**

**{**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**

**}**

**public \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ toString()**

**{**

**return "L'attribut valeur est de " + valeur + ", somme = " + somme +**

**", nombre d'objets créés = " + nbObjets;**

**}**

**} // fin de la classe Exemple**

Complétez maintenant l'application **TestExemple** :

|  |
| --- |
| **public class TestExemple**  **{**  **public static void main(String args[])**  **{**  **Exemple obj1 = new Exemple(5);**  **System.out.println("objet 1 : " + obj1);**  **Exemple obj2 = new Exemple(10);**    **System.out.println("objet 1 : " + obj1);**  **System.out.println("objet 2 : " + obj2);**  **System.out.println("\nLa somme des attributs valeur = " + \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);**  **System.out.println("Le nombre d'objets Exemple = " +**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);**  **}**  **}** |

Quels seront les résultats affichés lors de l'exécution de la classe **TestExemple** ?

10. Complétez la classe **CompteEpargne** qui utilise une variable de classe privée pour stocker le *tauxAnnuelInteret* à 3.5% de tous les détenteurs d’un compte. Chaque objet de la classe contient la variable d’instance privée *soldeCompte* qui indique le montant dont l’épargnant dispose sur son compte. Fournissez les constructeurs (celui sans paramètre et celui avec paramètres), les méthodes *set*, les méthodes *get*, la méthode *toString* qui retourne le solde du compte dans un format monétaire et la méthode *calculerInteretMensuel* qui calcule le nouveau *soldeCompte* en lui ajoutant le résultat de la multiplication de sa valeur avec le *tauxAnnuelInteret*, divisé par 12.

Complétez le programme pilote qui teste la classe **CompteEpargne** en instanciant deux objets **CompteEpargne**, *epargnant1* et *epargnant2*, ajustés respectivement à 2000 et 3000. Par la suite, modifiez le *tauxAnnuelInteret*à 4%, calculez et affichez les nouveaux soldes. Calculez et affichez les soldes du mois suivant avec un *tauxAnnuelInteret* de 5%.

**import java.text.\*;**

**public class Compte**

**E**

**pargne**

**{**

**}**

**public class TestCompte**

**E**

**pargne**

**{**

**public static void main(String[] args)**

**{**

***// avec taux d'intérêt à 4%***

***// avec taux d'intérêt à 5%***

**}**

**}**

**Résultats obtenus :**

**Solde de l'épargnant 1 : 2006,67 $**

**Solde de l'épargnant 2 : 3010,00 $**

**Solde de l'épargnant 1 : 2015,03 $**

**Solde de l'épargnant 2 : 3022,54 $**