|  |  |
| --- | --- |
| SLAM2 | Révisions Programmation objet |
| Questionnaire 3 | Java : Classes et objets |

1. Carte joker = new Carte("Valet", "Trèfle");

|  |  |
| --- | --- |
| x | joker est une variable |

|  |  |
| --- | --- |
| x | joker est un objet |

|  |  |
| --- | --- |
| x | Carte est le nom d'une classe |

1. Client unClient = new Client(1678, "Jean", "Dupont");

|  |  |
| --- | --- |
|  | unClient est une classe |

|  |  |
| --- | --- |
| x | unClient est un objet |

|  |  |
| --- | --- |
| x | unClient est une variable |

|  |  |
| --- | --- |
| x | new permet de créer un nouvel objet |

|  |  |
| --- | --- |
| x | new est un opérateur |

1. Chien unChien = new Chien("Médor");

unChien.Aboyer();

string nom = unChien.getNom();

|  |  |
| --- | --- |
| x | Chien est un type de donnée |

|  |  |
| --- | --- |
| x | Chien est une classe |

|  |  |
| --- | --- |
|  | Aboyer est un membre privé |

|  |  |
| --- | --- |
|  | Aboyer est un champ de la classe |

|  |  |
| --- | --- |
| x | getNom est un membre public |

|  |  |
| --- | --- |
|  | Médor est un objet |

1. Un objet :

|  |  |
| --- | --- |
|  | Est un membre privé d'une classe |

|  |  |
| --- | --- |
| x | Est une instance de classe |

1. Quels identificateurs d'objets respectent nos normes de développement ?

|  |  |
| --- | --- |
| x | agence |

|  |  |
| --- | --- |
|  | Outil |

|  |  |
| --- | --- |
| x | unClient |

1. Soit la classe Voiture (- : private, + : public) :

|  |
| --- |
| **Voiture** |
| - String immatriculation  - int kilometres  - int nbPortes |
| + Voiture(string pImmat, int pNbKm, int pNbPortes)  + int getKilometrage()  + void Demarrer() |

a) Ecrire les instructions suivantes (on suppose que ces instructions sont dans le programme principal (Main) ) :

|  |
| --- |
| // créer un objet nommé uneVoiture, représentant une voiture 5 portes immatriculée 608DEF57 avec 45000 km  Voiture uneVoiture = new Voiture(« 608DEF57 », 45000, 5) ; |

|  |
| --- |
| // afficher le kilométrage de la voiture uneVoiture  System.out.println(uneVoiture.getKilometrage()) ; |

|  |
| --- |
| // démarer la voiture *uneVoiture*  uneVoiture.Demarrer() ; |

|  |
| --- |
| // déclarer un tableau de 20 voitures nommé *tbVoitures*  Voiture[] tbVoitures = new Voiture[20] ; |

|  |
| --- |
| // afficher le kilométrage de la première voiture du tableau  System.out.println(tbVoitures[0].getKilometrage()) ; |

b) Cochez les instructions erronées, on suppose qu'on dispose de *uneVoiture* et de *tbVoitures* (tableau plein):

|  |  |
| --- | --- |
| x | Voiture maVoiture = new("607ZER57", 450); |

|  |  |
| --- | --- |
| x | System.out.println(uneVoiture.nbPortes); |

|  |  |
| --- | --- |
|  | System.out.println(uneVoiture.getKilometrage()); |

|  |  |
| --- | --- |
| x | System.out.println(uneVoiture.Demarrer()); |

|  |  |
| --- | --- |
| x | System.out.println(tbVoitures.getKilometrage()); |

|  |  |
| --- | --- |
| x | System.out.println(tbVoitures); |

|  |  |
| --- | --- |
|  | uneVoiture.Demarrer(); |

|  |  |
| --- | --- |
| x | Voiture taVoiture = new("697YAB57", 1450, "5"); |

|  |  |
| --- | --- |
|  | System.out.println(tbVoitures[0].getKilometrage()); |

|  |  |
| --- | --- |
| x | for (Voiture : tbVoitures) {  System.out.println(uneVoiture.getKilometrage());  } |

|  |  |
| --- | --- |
|  | int i = 0;  int nbKM = 0;  do {  nbKM += tbVoitures[i].getKilometrage();  i++;  } while (i < 5); |

|  |  |
| --- | --- |
| x | int nbKM = 0;  do {  nbKM += tbVoitures[i].getKilometrage();  } while (nbKM < 5000); |

|  |  |
| --- | --- |
| x | int i = 0;  while (tbVoitures[i].getKilometrage() < 2000) {  System.out.println(Kilometrage);  } |

|  |  |
| --- | --- |
| x | int i = 0;  while (tbVoitures[i].getKilometrage() < 2000 && i < 20) {  System.out.println(tbVoitures.getKilometrage());  } |

|  |  |
| --- | --- |
| x | int i = 0;  while (i < 20 && tbVoitures[i].getKilometrage() < 2000) {  System.out.println(tbVoitures[i].getKilometrage());  } |

|  |  |
| --- | --- |
|  | int i = 0;  while (i < 20 && tbVoitures[i].getKilometrage() < 2000) {  System.out.println(tbVoitures[i].getKilometrage());  i++;  } |