



**Module : IDAI-54 Programmation Orientée Objet en C++/Java**

**Devoir**

**Botaina Ahadri**

**IDAI S5**

## **Exercice :**

Vous devez concevoir une classe appelée Voiture qui représente les caractéristiques d'une voiture et ses comportements associés. La classe doit inclure les éléments suivants :

### **Attributs :**

1. marque (de type `std::string`) : la marque de la voiture.
2. modele (de type `std::string`) : le modèle de la voiture.
3. annee (de type `int`) : l'année de fabrication de la voiture.
4. kilometrage (de type `float`) : le kilométrage actuel de la voiture.
5. vitesse (de type `float`) : la vitesse actuelle de la voiture.

### **Méthodes :**

1. Constructeur par défaut : Initialise tous les attributs avec des valeurs par défaut.
2. Constructeur avec paramètres : Permet d'initialiser la voiture avec une marque, un modèle, une année, un kilométrage et une vitesse initiale.
3. Méthode `accelerer(float valeur)` : Incrémente la vitesse actuelle de la voiture en fonction de la valeur passée en paramètre.
4. Méthode `freiner(float valeur)` : Diminue la vitesse actuelle de la voiture en fonction de la valeur passée en paramètre, sans que la vitesse ne devienne négative.
5. Méthode `afficherInfo()` : Affiche les informations sur la voiture (marque, modèle, année, kilométrage, vitesse).
6. Méthode `avancer(float distance)` : Incrémente le kilométrage en fonction de la distance passée en paramètre.
7. Destructeur : Affiche un message indiquant que la voiture est détruite.

## Solution:

```
#include <iostream>
```

```
#include <string>
```

```
using namespace std;
```

```
class Voiture {
```

```
private:
```

```
    string marque;
```

```
    string modele;
```

```
    int annee;
```

```
    float kilometrage;
```

```
    float vitesse;
```

```
public:
```

```
    Voiture() : marque( "indefini" ), modele( "indefini" ), annee(1000), kilometrage(0.0), vitesse(0.0) {}
```

```
    Voiture(string marque, string modele, int annee, float kilometrage, float vitesse)
```

```
        : marque(marque), modele(modele), annee(annee), kilometrage(kilometrage), vitesse(vitesse) {}
```

```
    void accelerer(float valeur) {
```

```
        vitesse += valeur;
```

```
}
```

```
    void freiner(float valeur) {
```

```
        if (vitesse >= valeur) {
```

```
            vitesse -= valeur;
```

```
        } else {
```

```
            vitesse = 0;
```

```
        }
```

```
}
```

```

void afficherInfo() {
    cout << "La marque de la voiture est : " << marque << endl;
    cout << "Le modele de la voiture est : " << modele << endl;
    cout << "L'annee de fabrication de la voiture est : " << annee << endl;
    cout << "Le kilometrage actuel de la voiture : " << kilometrage << " km" << endl;
    cout << "La vitesse actuelle de la voiture : " << vitesse << " km/h" << endl;
}

void avancer(float distance) {
    kilometrage += distance;
}

~Voiture() {
    cout << "La voiture de marque " << marque << " est detruite." << endl;
}

};

int main() {
    string marque, modele;
    int annee;
    float kilometrage, vitesse, valeurAccer, distance, valeurFrein;

    voiture.afficherInfo();

    cout << "Donner la marque de la voiture" << endl;
    cin >> marque;
    cout << "Donner le modèle de la voiture" << endl;
    cin >> modele;
    cout << "Donner l'annee de fabrication de la voiture" << endl;
    cin >> annee;

```

```
cout << "Donner le kilometrage de la voiture" << endl;

cin >> kilometrage;

cout << "Donner la vitesse actuelle de la voiture" << endl;

cin >> vitesse;


Voiture voiture(marque, modele, annee, kilometrage, vitesse);
voiture.afficherInfo();


cout << "Donner la valeur d'acceleration" << endl;

cin >> valeurAccer;

voiture.accelerer(valeurAccer);


cout << "Donner la distance" << endl;

cin >> distance;

voiture.avancer(distance);


cout << "Donner la valeur de freinage" << endl;

cin >> valeurFrein;

voiture.freiner(valeurFrein);


voiture.afficherInfo();

return 0;

}
```