

1. Введіть відсутні ключове слово, щоб оголосити констатну змінну з назвою "var" типу double.

```
_____ double ____ = 3.4;
```

2. Оголошіть об'єкт з назвою "st" типу Student, а потім зробіть виклик його функції printAge ().

```
_____ st;  
st. _____;
```

3. Введіть відсутні ключові слова, щоб оголосити констатну функцію члена printAge () для класу Student.

```
_____ Student {  
public:  
void printAge() _____ ;  
};
```

4. Константні функції-елементи

- ... не можуть бути викликані
- ... не можуть змінювати будь-який не константний елемент
- ... можуть змінювати не константні дані

5. Константна змінна ...

- ... не можливо змінити
- ... можливо змінити
- ... не може бути ініціалізованою

6. У вас є клас "Student" з двома членами, "age" і "num". Заповніть пробіли для ініціалізації цих членів у списку ініціалізатора конструктора з відповідними значеннями.

```
Student::Student(int a, int b)  
_ age(a),  
_ _____ (b) {  
}
```

7. Ініціалізуйте елементи у списку ініціалізації конструктора, а далі виведіть їх у тілі конструктора

```
Student::Student(int a, int b)
: regVar(a),
  constVar(b)
{ cout << _____ <<
  constVar _____ endl;
}
```

8. Константна змінна-член ...

- ... може бути ініціалізована де завгодно
- ... повинна бути ініціалізована у списку ініціалізації конструктора
- ... повинна бути ініціалізована в тілі конструктора

9. Заповніть пробіли, щоб ініціалізувати членів класу "Birthday" у списку ініціалізаторів конструктора.

```
Birthday::Birthday(int m, int d, int y)
: month(m),
  day(d),
  year(_____)
{
}
```

10. Необхідно створити клас "People" з трьома приватними цілими членами: birthMonth, BirthDay, BirthYear.

```
class _____ {
public:
    People();

    _____
    int _____;
    int birthDay;
    _____ birthYear;
};
```

11. Необхідно оголосити клас "People" з двома приватними членами: name типу "string" та dateOfBirth типу "Birthday". Не забудьте включити файл заголовка "Birthday.h" і файл заголовка для типу string.

```
#include <_____>
#include "_____"
class _____ {
public:
    People(string n, Birthday bo);
private:
    string name;
    _____ dateOfBirth;
};
```

10. Необхідно оголосити конструктор "People", який приймає два параметри та ініціалізувати його приватні елементи: name та dataOfBirth.

```
People::People(string x, Birthday bo)
    _name( _ ), dateOfBirth(bo)
{
}
```

11. Необхідно визначити функцію printInfo (), яка друкує name та birthdate, використовуючи функцію dateOfBirth printDate ().

```
void People:: _____ () {
    cout << name << endl;
    _____ . _____ ();
}
```

12. Оголосіть об'єкт типу "People", який приймає рядок як перший параметр для його конструктора, та об'єкт "Birthday" - як другий. Оголосіть об'єкт birthObj щоб передати його конструктору "People".

```
_____ birthObj(10, 30, 1989);
People john("John", _____);
```