Тест по Ruby

1. Выберите корректные формы записи строк.
   1. "текст"
   2. 'текст'
   3. %{2' "текст" с кавычками}
   4. "текст с двойными кавычками (")"
   5. "текст с двойными кавычками \""
2. Выберите корректные способы объединения строк (`var1` и `var2` — строковые переменные).
   1. "#{var1}#{var2}"
   2. '#{var1}#{var2}'
   3. var1 + var2
   4. var1 << var2
   5. var1 >> var2
3. Выберите корректные формы записи чисел.
   1. 42
   2. 31\_337
   3. 12.34
   4. 1234e-2
   5. 0x2a
4. Выберите корректные формы записи массивов
   1. ['a', 'b', 'c']
   2. %w(a b c)
   3. %w[a b c]
   4. %a{a b c}
5. Выберите форму записи хэша идентичную `{a: 1, b: 2}`.
   1. { :a => 1, :b => 2 }
   2. { 'a' => 1, 'b' => 2 }
   3. { a => 1, b => 2 }
   4. ни одна из них
6. Выберите все falsy-значения ruby (те которые не приводят к выполнению при подстановке в if).
   1. false
   2. 0
   3. nil
   4. пустая строка
7. Поддерживают ли ruby множественное наследование?
   1. да
   2. нет
8. Что означает следующая запись `Class.new(A)`?
   1. инициализация класса A
   2. создание анонимного класса отнаследованного от класса A
   3. код выдаст ошибку
9. Выберите корректные утверждения
   1. Классы не могут содержать другие классы и модули внутри собственного неймспейса
   2. Модули не могут содержать другие классы и модули внутри собственного неймспейса
   3. Модули не могут иметь собственные объекты
   4. Классы нельзя включать (include/extend) в другие классы и методы
10. Для чего предназначены методы `attr\_accessor`, `attr\_reader`, `attr\_writer`.
    1. для создания методов предоставляющих доступ к instance-переменным
    2. для инициализации instance-переменных
    3. для фильтрации входных параметров конструктора
11. Укажите что делает код `A.include(B)` при условии что A - класс, а B — модуль.
    1. расширяет instance-методы класса A instance-методами определенными в модуле B
    2. расширяет class-методы класса A instance-методами определенными в модуле B
    3. код выдает ошибку
    4. ничего из приведенного
12. Укажите что делает код `A.extend(B)` при условии что A - класс, а B — модуль.
    1. расширяет instance-методы класса A instance-методами определенными в модуле B
    2. расширяет class-методы класса A instance-методами определенными в модуле B
    3. код выдает ошибку
    4. ничего из приведенного
13. Для чего применяется ключевое слово yield
    1. для обращения к текущему классу
    2. для вызова исключения
    3. для вызова блока переданного в метод в качестве аргумента
    4. ни для чего из перечисленного
14. Какое значение **вернет** код `-> { 1; return 2; 3; }.call`?
    1. 1
    2. 2
    3. 3
    4. вызовет исключение
15. Какое значение **вернет** код `Proc.new { 1; return 2; 3; }.call`?
    1. 1
    2. 2
    3. 3
    4. вызовет исключение