Питання ООП

1. Основні засади об'єктного підходу. Типізація.
2. Основні засади об'єктного підходу. Модульність.
3. Еволюція методологій програмування. Парадигми програмування.
4. Що таке ООП?
5. Навіщо використовувати ООП?
6. Переваги Об’єктно-орієнтованого підходу над процедурним в програмуванні.
7. Об'єкт із погляду ООП. Стан. Поведінка.
8. Об'єкт із погляду ООП. Ідентичність та життєвий цикл об'єктів.
9. Об'єкт із погляду ООП. Відношення між об'єктами Що таке клас?
10. Що таке область видимості змінної в мовах ООП?
11. Наведіть приклад області видимості змінних в різних мовах ООП і порівняйте їх.
12. У чому різниця між класом та структурою?
13. Чи можна викликати метод базового класу без створення екземпляра?
14. Що таке спадкування?
15. Які бувають типи спадкування?
16. Що таке гібридне спадкування?
17. Що таке ієрархічне спадкування?
18. Які обмеження спадкування?
19. Що таке підклас?
20. Що таке поліморфізм?
21. Що таке перевантаження методу?
22. Що таке перевизначення методу?
23. Що таке перевантаження оператора?
24. Що таке інкапсуляція?
25. Що таке «специфікатори доступу»?
26. Для чого потрібен модифікатор доступу “public”?
27. Для чого потрібен модифікатор доступу “private”?
28. Для чого потрібен модифікатор доступу “protected”?
29. Який модифікатор доступу встановлюється за замовчуванням?
30. У чому різниця між модифікаторами публічного, приватного та захищеного доступу?
31. Що таке абстракція даних?
32. Як домогтися абстракції даних?
33. Чи можна створити екземпляр абстрактного класу?
34. Що таке інтерфейс?
35. Що таке конструктор?
36. Що таке деструктор?
37. Види конструкторів.
38. Що таке garbage collector?
39. Що таке кінцева змінна?
40. Що таке виняток?
41. Що таке обробка винятків?
42. У чому різниця між помилкою та винятком?
43. Що таке блок "try"?
44. Що таке блок “catch”?
45. Які обмеження ООП?
46. Що таке Getter?
47. Наведить приклад створення методу “Getter” в мовах ООП.
48. Що таке Setter?
49. Наведить приклад створення методу “Setter” в мовах ООП.
50. Навіщо потрібні Getter/Setter?
51. Які модифікації рівня доступу ви знаєте?
52. Дайте визначення поняттю “метод”.
53. Навіщо потрібен оператор instanceof?
54. Навіщо потрібні ініціалізації?
55. Які бувають блоки ініціалізації?
56. Який порядок виклику конструкторів та блоків ініціалізації двох класів: нащадка та його предка?
57. Навіщо потрібні класи?
58. Навіщо потрібні об’єкти?
59. Чим відрізняється екземпляр класу від об'єкту класу?
60. Наведіть приклад синтаксту наслідування в мовах ООП.
61. Навіщо потрібний явний конструктор?
62. Наведіть приклад створення явного конструктора в мовах ООП.
63. Наведіть приклад створення об’єкту класу і передачі аргументів в конструктор в мовах ООП.
64. Що означає “Синтаксичний цукор”?
65. Чим відрізняється конструкція Синтаксичного цукру від конструкції класу в мові JS?
66. Що таке “Ромбоподібне успадкування(«diamond problem»)”?
67. Вкажіть механізм затримок при роботі з ромбовидним успадкуванням.
68. Вкажіть приклад обробки проблеми ромбовидного успадкування в мовах ООП.
69. Що таке Життєвий Цикл Розробки Системи?
70. З чого складається етап Планування системи?
71. З чого складається етап Аналізу системи?
72. З чого складається етап Дизайну системи?
73. З чого складається етап Розробки системи?
74. З чого складається етап Впровадження та Розгортання системи?
75. З чого складається етап Інтеграції системи?
76. З чого складається етап Підтримка системи?
77. Навіщо використовується імпорт класів?
78. Наведіть приклади імпорту класів в мовах ООП.
79. Що означає абревіатура “DRY”?
80. Що таке Віртуальна машина?
81. Для чого потрібна Віртуальна машина?
82. Назвіть переваги і недоліки користування Віртуальною машиною.