

# Zadania - c1

1. Napisz skrypt PHP, który wyświetli aktualną datę i czas.
2. Utwórz skrypt PHP, który wczyta z linii poleceń imię użytkownika i przywita go po imieniu.
3. Napisz skrypt PHP, który obliczy sumę dwóch podanych z linii poleceń liczb.
4. Utwórz skrypt PHP, który konwertuje kilometry na mile. Liczbę kilometrów podaj z linii poleceń.
5. Napisz skrypt PHP, który z linii poleceń otrzyma listę liczb i wyświetli największą z nich.
6. Utwórz skrypt PHP, który wczyta z linii poleceń ciąg znaków i wyświetli liczbę znaków w ciągu.
7. Napisz skrypt PHP, który dla podanej z linii poleceń liczby wyświetli jej tabliczkę mnożenia do 10.
8. Utwórz skrypt PHP, który sprawdzi, czy podana z linii poleceń liczba jest liczbą pierwszą.
9. Napisz skrypt PHP, który przeliczy stopnie Fahrenheit na Celsjusza. Stopnie Fahrenheit podaj z linii poleceń.
10. Napisz funkcję PHP, która sprawdzi, czy dany rok jest rokiem przestępnym.
11. Utwórz funkcję PHP, która obliczy silnię podanej liczby.
12. Napisz skrypt PHP, który zamieni wszystkie spacje w podanym ciągu znaków na podkreślenia.
13. Utwórz funkcję PHP, która porówna dwie tablice i zwróci elementy, które się różnią.
14. Napisz funkcję PHP, która zwróci n-ty element ciągu Fibonacciego.
15. Utwórz funkcję PHP, która sprawdzi, czy podany ciąg znaków jest anagramem innego ciągu znaków.
16. Napisz funkcję PHP, która obliczy sumę wszystkich liczb parzystych w podanym zakresie.
17. Utwórz funkcję PHP, która zwróci odwróconą wersję podanego ciągu znaków bez użycia funkcji `strrev()`.
18. Napisz skrypt PHP, który dla podanej listy liczb zwróci listę zawierającą tylko liczby pierwsze.
19. Utwórz funkcję PHP, która zwróci `true`, jeśli podany ciąg znaków jest palindromem, a `false` w przeciwnym przypadku.
20. Utwórz klasę `Car` z właściwościami takimi jak marka, model i rok produkcji oraz metodą wyświetlającą pełne informacje o samochodzie.
21. Zaimplementuj klasę `Calculator` z metodami do podstawowych operacji matematycznych (dodawanie, odejmowanie, mnożenie, dzielenie).

22. Stwórz klasę `Person` z właściwościami `imię`, `nazwisko` i `wiek` oraz metodą wyświetlającą powitanie z imieniem i nazwiskiem.
23. Napisz klasę `BankAccount` z metodami do wpłacania i wypłacania pieniędzy oraz sprawdzania aktualnego stanu konta.
24. Utwórz klasę `Book` z właściwościami `tytuł`, `autor` i `cena` oraz metodą wyświetlającą pełne informacje o książce.
25. Utwórz klasę bazową `Vehicle` z właściwościami takimi jak `marka`, `model` i `rok produkcji` oraz metodą `displayInfo()` wyświetlającą informacje o pojeździe. Następnie zaimplementuj klasy dziedziczące `Car` i `Motorcycle`, dodając specyficzne właściwości i metody dla każdego typu pojazdu.
26. Stwórz klasę `Person` z podstawowymi informacjami takimi jak `imię` i `nazwisko`. Następnie utwórz klasy `Student` i `Teacher` dziedziczące po `Person`, dodając specyficzne właściwości i metody, takie jak numer indeksu dla studenta i przedmiot nauczania dla nauczyciela.
27. Zaimplementuj klasę `Shape` z metodą abstrakcyjną `calculateArea()`. Utwórz klasy `Circle`, `Rectangle` i `Triangle` dziedziczące po `Shape` i implementujące metodę `calculateArea()` w sposób odpowiedni dla każdego kształtu.
28. Napisz klasę `Animal` zawierającą podstawowe cechy i zachowania zwierząt, takie jak `nazwa` i metoda `makeSound()`. Stwórz klasy dziedziczące `Dog` i `Cat`, które nadpisują metodę `makeSound()`, by wydawać odpowiednie dźwięki dla psa i kota.
29. Utwórz klasę `ElectronicDevice` z podstawowymi informacjami o urządzeniu elektronicznym oraz metodami `turnOn()` i `turnOff()`. Następnie zaimplementuj klasy `Smartphone` i `Laptop` dziedziczące po `ElectronicDevice`, dodając unikalne właściwości i metody, np. `installApp()` dla smartfonu.
30. Utwórz tablicę zawierającą nazwy dni tygodnia, a następnie napisz funkcję, która zwróci nazwę dnia na podstawie podanego numeru dnia (gdzie 1 = poniedziałek, 7 = niedziela).
31. Stwórz tablicę przechowującą informacje o książkach (`tytuł`, `autor`, `rok wydania`), a następnie napisz funkcję, która wyświetli te informacje w formacie: "Tytuł: Autor (Rok wydania)".
32. Napisz funkcję, która przyjmuje tablicę liczb i zwraca nową tablicę zawierającą tylko liczby parzyste z oryginalnej tablicy.
33. Utwórz tablicę wielowymiarową reprezentującą macierz 3x3, a następnie napisz funkcję, która obliczy i zwróci sumę elementów na głównej przekątnej macierzy.
34. Napisz funkcję, która przyjmuje tablicę słów, a następnie zwraca tę tablicę posortowaną alfabetycznie, ignorując wielkość liter.
35. Utwórz klasę `BankAccount` z prywatnymi właściwościami `accountNumber` i `balance`. Dodaj publiczne metody do ustawiania i pobierania stanu konta oraz do wpłat i wypłat, zapewniając, że saldo nie może spaść poniżej zera.
36. Zaimplementuj klasę `Person` z prywatnymi właściwościami `name`, `age` i `email`. Utwórz publiczne metody `setName()`, `setAge()`, `setEmail()` do ustawiania wartości oraz odpowiednie metody do ich pobierania, z walidacją danych wejściowych (np. wiek jako dodatnia liczba, email z odpowiednim formatem).
37. Stwórz klasę `Car` z prywatnymi właściwościami `make`, `model` i `year` oraz metodą publiczną `displayCarInfo()`, która zwraca informacje o samochodzie. Dodaj konstruktor

do inicjalizacji tych właściwości oraz publiczne metody do ich modyfikacji z odpowiednimi ograniczeniami (np. rok produkcji nie może być w przyszłości).

38. Projektując klasę `Product`, zdefiniuj prywatne właściwości `name`, `price` i `quantity` oraz metodę `calculateTotalValue()`, która zwróci całkowitą wartość produktu na podstawie ceny i ilości. Zapewnij publiczne metody do ustawiania i pobierania tych właściwości, z odpowiednimi warunkami (np. cena i ilość muszą być większe od zera).
39. Zaimplementuj klasę `MathUtils` z metodą statyczną `calculateCircleArea($radius)`, która oblicza i zwraca pole koła na podstawie podanego promienia.
40. Utwórz klasę `StringUtils` z metodą statyczną `reverseString($string)`, która przyjmuje ciąg znaków jako argument i zwraca odwróconą wersję tego ciągu.
41. Stwórz klasę `TemperatureConverter` z metodami statycznymi `celsiusToFahrenheit($celsius)` i `fahrenheitToCelsius($fahrenheit)`, które przeliczają temperaturę między stopniami Celsjusza a Fahrenheita.
42. Napisz klasę `Validator` z metodą statyczną `isEmailValid($email)`, która sprawdza, czy podany ciąg znaków jest poprawnym adresem email i zwraca wartość `true` lub `false`.
43. Utwórz klasę `ConfigLoader` z prywatną statyczną właściwością przechowującą konfigurację aplikacji jako tablicę oraz publiczną metodą statyczną `getConfig($key)`, która zwraca wartość konfiguracji dla podanego klucza, jeśli istnieje, lub `null` w przeciwnym razie.