

- I. Utwórz klasę `Drzewo` z polami `boolean wiecznieZielone`, `int wysokosc` oraz `String przekrojDrzewa`, następnie zaimplementuj konstruktory inicjujące pola i metodę `String toString()` zwracającą informacje o drzewie.
- II. Utwórz klasę `DrzewoIglaste` dziedziczącą po klasie `Drzewo`, oraz definiującą pola `int iloscIgiel` i `double dlugoscSzyszki`. Następnie przygotuj konstruktor inicjujący wszystkie pola oraz metodę `String toString()` zwracającą informacje o drzewie wraz z wartościami pól opisujących drzewo iglaste.

Utwórz klasę `DrzewoLisciaste` dziedziczącą po klasie `Drzewo`, oraz definiującą pole `int ksztaltLiscia` i metodę `String toString()` zwracającą informacje o drzewie.

Utwórz klasę `DrzewoOwocowe` dziedziczącą po klasie `DrzewoLisciaste`, oraz definiującą pole `String nazwaOwoca`, podobnie jak w poprzednich zadaniach uzupełnij klasę o konstruktor i funkcjonalną metodę `toString`.
- III. Wykorzystując kod programów z poprzednich ćwiczeń utwórz program, który wypełni tablicę `Drzewo[]` las drzewami takimi jak: *sosny*, *modrzewie*, *dęby*, *osiki*, *morelowce* czy *śliwy*. Wykorzystując mechanizm polimorfizmu wyświetl wszystkie informacje o wszystkich drzewach.

Na koniec zaimplementuj we wszystkich klasach metodę `zerwijOwoc`. Jeżeli drzewo nie będzie drzewem owocowym, przez co nie będzie miało owoców - podniesiony zostanie wyjątek `DrzewoBezOwocoweException`.
- IV. Utwórz klasę `Alarm` dziedziczącą po klasie `Exception`. Następnie przygotuj klasę `DetektorDymu` zawierającą metodę `sprawdz`, która może podnieść wyjątek `Alarm` jeśli wyczuje dym.
- V. Utwórz klasę `Rakieta`, zbudowaną z pól: `String nazwa` i `int wagaPaliwa`, konstruktora inicjującego pola oraz następujących metod: `zatankuj` - ładującą losową ilość paliwa, oraz `start` - rozpoczynającą procedurę startową. W ramach kontroli przedstartowej metoda `start` sprawdzi stan paliwa i jeżeli jest on mniejszy niż 1000, podniesiony zostanie wyjątek `Exception` z komentarzem 'start anulowany - za mało paliwa'.
- VI. Wykorzystując klasę `FileInputStream` odczytaj zawartość dowolnego pliku tekstowego i podaj z jakich znaków składa się na ten plik i jaka jest liczba wystąpień każdego z nich.