

## 1 Projekt

Utwórz projekt Shapes w środowisku Netbeans z plikami, które znajdują się w dołączonym archiwum *shapes.zip*. Wszystkie klasy powinny znajdować się w pakiecie **shapes**.

## 2 Klasy

Analiza, testowanie i modyfikacja klas specyfikujących klasę **Shape** (Kształt).

Zaimplementowano dwie klasy **Square** i **Circle**. Dla każdej z tych klas jesteśmy zainteresowani otrzymaniem pola powierzchni figury (zauważ, że metoda **area** jest dziedziczona z interfejsu **Shape**).

1. Przetestuj **Square** oraz **Circle** - z wykorzystaniem JUnit 3.
2. Zaimplementuj klasę **Triangle** (trójkąt), która również będzie dziedzyczyła z **Shape**. Klasa **Triangle** powinna mieć ten sam zestaw metod co pozostałe klasy dziedziczące. Klasa powinna być inicjalizowana wartościami długości podstawy oraz wysokości na nią opuszczonej (**base** oraz **height**).
3. Pamiętaj o testowaniu każdej metody.

## 3 Kolekcje

Być może wygodniejszą formą pracy z kształtami będzie zgromadzenie ich w zbiorze **ShapeSet**. Klasa **ShapeSet** zarządza zbiorem obiektów typu **Shape**.

1. Zastanów się jak chcesz przechowywać kształty? Co będzie odpowiednie **set**, **list**, **queue** czy może **map**. Jakie są zalety i wady?
2. Utwórz metodę **addShape** dodającą kształt do zbioru. Zapewnij, że żadne dwa te same kształty nie znajdują się w zbiorze.
3. Zaimplementuj metodę **iterator** zwracającą iterator do zbioru elementów.
4. Uzupełnij metodę **toString**.
5. Pamiętaj o testowaniu każdej metody.

---

<sup>1</sup>Na podstawie kursów MIT OCW

## 4 Wyszukiwanie i plik

Teraz gdy mamy implementację klas reprezentujących kształty oraz zbior kształtów należy zaimplementować dwie metody w klasie **Shapes** (nie pomylić z klasą **Shape**).

1. **findLargest** - należy znaleźć figury o największym polu i utworzyć listę a następnie przekazać ją jako wartość zwracaną.
2. **readShapesFromFile** - Aby ułatwić sobie tworzenie obiektów do projektu dołączony jest plik z deklaracją kształtów. Należy go wczytać i utworzyć obiekt **ShapeSet** zawierający kształty występujące w pliku. (rozwiązanie należy wyszukać w dokumentacji lub przykładach w internecie — IO Javy zostanie dokładniej omówione na następnych zajęciach)
3. Pamiętaj o testowaniu każdej metody.