**Interfejs Interable** zawiera Iterator , który jest obiektem pozwalającym poruszać się (iterować) po innym obiekcie (kolekcji).

public interface Iterable<E>{

Iterator<E> iterator();

}

Klasy implementujące ten interfejs zobowiązują się do dostarczenia narzędzia pozwalającego na przeglądanie ich zawartości - iteratora.

interfejs Iterator<E> Oto pełna deklaracja:

public interface Iterator<E> {

boolean hasNext();

E next();

void remove();

}

Metoda hasNext informuje, czy w kolekcji są jeszcze elementy (co pozwala się upewnić, że można wywołać metodę next). Metoda next pozwala pobrać kolejny element z kolekcji, tzn. kolejne wywołania tej metody - aż do osiągnięcia stanu, gdy wywołanie metody hasNext da wartość false - przekażą wszystkie elementy kolekcji (każdy dokładnie raz). Nie ma w tym interfejsie metody pozwalającej wrócić na początek przeglądanej kolekcji. Jeśli będziemy chcieli obejrzeć kolekcję od początku, po prostu stworzymy nowy iterator!

Ostatnia metoda (remove) jest trochę innego rodzaju, jako jedyna modyfikuje kolekcję. Służy do usunięcia z niej elementu, który był jako ostatni przekazany jako wynik metody next. Z używaniem tej metody wiąże się wiele ograniczeń. Nie można jej wywołać przed pierwszym wywołaniem metody next, nie można jej także wywołać dwa razy pod rząd bez wywołania metody next. Co więcej, ta metoda jest opcjonalna (nie wszystkie implementacje tego interfejsu muszą potrafić usuwać).

**Interfejs Callable** to zwracanie wyników przez wątki. Cllaable jest interfejsem generycznym, zdolnym do zwracania dowolnego typu wyników.

Zwracanie wyników, ewentualnie sygnalizację i odbieranie wyjątków przez

zadania możemy realizować poprzez interfejs Callable. Obiekt Callable posiada wartość zwrotną. Interfejs Callable jest typem parametryzowanym zawierającym jedną metodę o nazwie call. Ma on następującą postać.

public interface Callable<V> {

V call() throws Exception;

}

Parametr typowy określa typ zwracanej wartości. Na przykład Interfejst Callable<integer> reprezentuje asynchroniczne działania, których wynikiem jest obiekt typu Integer.