

## Krzysztof Galek 249605

**Interfejs Iterable** jest jednym z głównych interfejsów interfejsu API kolekcji, zapewnia przejście przez elementy kolekcji oraz pobieranie lub usuwanie elementów kolekcji. Klasa implementująca interfejs Iterable, może być iterowana za pomocą pętli for-each. Interfejs Iterable implementuje się podczas projektowania niestandardowych struktur danych.

```
public class Przykład implements Iterable
{
    private List < Przykład > przykład = new ArrayList < Przykład > ();
    public Iterator < Przykład > iterator()
    {
        return this.p rzykład.iterator();
    }
    Przykład przykład = new Przykład;
    for( Przykład przykład :: Przykład)
    {
        ...
    }
```

**Interfejs Closeable** to przestarzały interfejs służący do zamykania zasobów dodany w Java Development Kit 5. Posiada ograniczenie, ponieważ może tylko generować wyjątek IOException, którego nie można zmienić bez zerwania kodu legacy. W wersji JDK7 wprowadzono interfejs AutoCloseable zwracający wyjątek. Interfejs Closeable służy jako rozszerzenie interfejsu AutoCloseable. Za pomocą konstrukcji "try-with-resources" można zamknąć wszelkie zasoby, które implementują Closeable lub AutoCloseable.

```
public interface Closeable extends AutoCloseable
{
    public void close() throws IOException;
```

```
}
```

```
public interface AutoCloseable
```

```
}
```

```
void close() throws Exception;
```

```
}
```