

## Podstawowe adnotacje JPA i Hibernate

Aby klasa była rozpoznawana jako encja, musi posiadać adnotację JPA – `Entity`. Jeżeli tabela bazy danych ma inną nazwę niż klasa encji, należy zastosować też adnotację `Table` z atrybutem `name`, który określi nazwę tabeli.

Przykładem może być klasa `User`:

```
import javax.persistence.Entity;
import javax.persistence.Table;
// ...
@Entity
@Table(name = "users")
public class User implements Serializable { // ...
```

Mapowanie kolumn jest nieco bardziej złożonym procesem. Po pierwsze należy zauważyć, że kolumny mogą zostać przyporządkowane nie tylko polom klasy, lecz także jej właściwościom. Właściwość (ang. *property*) to jedna lub dwie spośród metod:

```
Type getXXX()
    metoda pobierająca wartość właściwości (ang. getter),
void setXXX(Type value)
    metoda ustawiająca wartość właściwości (ang. setter).
```

Obie metody posiadają ten sam człon `xxx` – nazwę właściwości. Również typ właściwości (`Type`) jest w obu przypadkach ten sam. Więcej na ten temat postaram się napisać innym razem.

Adnotacjami dotyczącymi kolumn opatrzone jest pole albo *getter* właściwości. Najczęściej używana jest adnotacja `Column`, która zawiera atrybut `name`, wskazujący kolumnę tabeli mapowaną na daną składową klasy. Za przykład może posłużyć metoda `getId`:

```
import org.hibernate.validator.NotNull;
import javax.persistence.Column;
import javax.persistence.Id;
import javax.persistence.GeneratedValue;
// ...
@Id
@GeneratedValue
@NotNull
@Column(name = "id")
public Long getId() { // ...
```

Inne typowe adnotacje JPA, to:

```
Id
    oznacza, że pole lub właściwość jest kluczem głównym tabeli,
GeneratedValue
    oznacza, że wartość kolumny będzie generowana automatycznie (jeżeli nie zostanie
    sprecyzowana inna metoda, Seam pobiera tę wartość z sekwencji
    hibernate_sequence),
Version
```

kolumna oznaczona tą adnotacją zawiera wersję rekordu – operacja `insert` ustawia tę wartość na 0, a `update` zwiększa ją o 1.

Teraz pora na najpopularniejsze adnotacje Hibernate:

`NotNull`

oznacza, że kolumna nie może przyjmować wartości `NULL`,

`NotEmpty`

kolumna zawierająca tekst, która nie może przyjmować wartości `NULL` ani pustego łańcucha znaków,

`Length`

określa minimalną (atrybut `min`) oraz maksymalną (`max`) długość tekstu,

`Min`

określa minimalną wartość liczbową,

`Max`

określa maksymalną wartość liczbową,

`Range`

określa minimalną (atrybut `min`) oraz maksymalną (`max`) wartość liczbową.