

Аннотация
`@Transient`

@Transient

Помечает те поля в сущности, которые мы не хотим связывать с таблицей в БД

- Поле, помеченное аннотацией `@Transient` не будет сохраняться в БД и Hibernate вообще не будет его "замечать".
- Полезно использовать для динамически вычисляемых полей

Пример использования на следующем слайде...

```
@Entity
```

```
@Table(name = "Triangle")
```

```
public class Triangle {
```

```
    @Id
```

```
    @Column(name = "id")
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private int id;
```

```
    // Эти поля храним в таблице в БД
```

```
    @Column(name = "a")
```

```
    private int a;
```

```
    @Column(name = "b")
```

```
    private int b;
```

```
    @Column(name = "c")
```

```
    private int c;
```

```
    // Это поле не храним в таблице в БД (не будет такой колонки даже)
```

```
    // Зачем хранить периметр в БД, если его можно быстро вычислить динамически на основании
```

```
    // имеющихся данных
```

```
    @Transient
```

```
    private int perimeter;
```

```
    // Вычислили прямо в программе и вернули
```

```
    public int getPerimeter() {
```

```
        return this.a + this.b + this.c;
```

```
    }
```

```
}
```

Сущность "Треугольник"

Длины сторон a, b, c храним в БД

Периметр не храним в БД