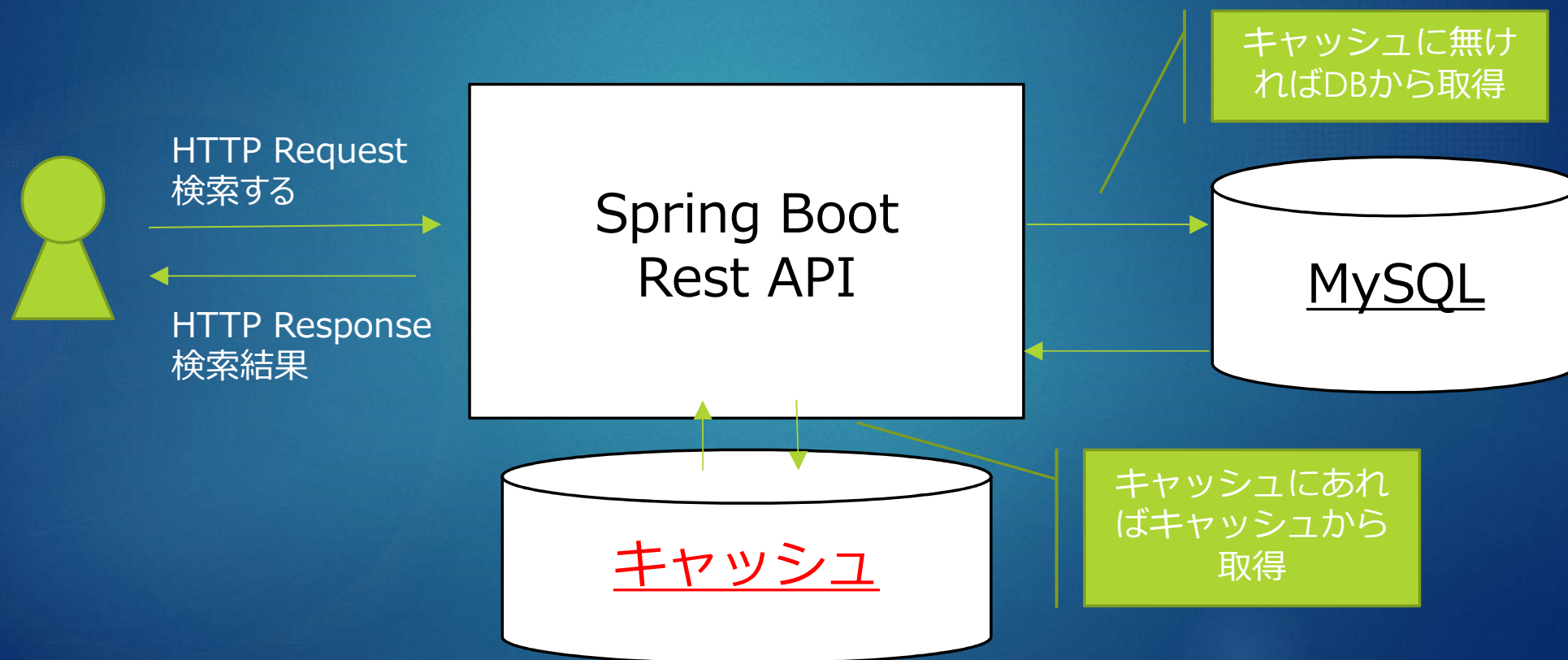


# キャッシュを使用する

検索系のAPIにて、毎回データベースに問い合わせるのではなく、キャッシュを活用して効率的にデータを問い合わせる方法を確認します。また、データの新規作成・更新・削除を行った時には、保持しているキャッシュと実データ間で不整合が生じてしまいますので、適切にキャッシュクリアを行う事が重要です。



# キャッシュを使用する際に必要な依存関係

「Spring cache abstraction」を依存関係に追加します。

## 必要な依存関係



## pom.xml

```
37         <scope>test</scope>
38     </dependency>
39     <dependency>
40         <groupId>org.springframework.boot</groupId>
41         <artifactId>spring-boot-starter-cache</artifactId>
42     </dependency>
43 </dependencies>
44
```



# キャッシュを有効にする

「@EnableCaching」でキャッシュの設定をONにし、キャッシュ管理を行う検索系のメソッドに@Cacheableを付与します。

- ▶ @EnableCachingを@SpringBootApplicationのクラスに指定します

```
@SpringBootApplication↓  
@EnableCaching  
public class Spring3itemApplication
```

- ▶ @Cacheableを取得のメソッドに付与すると、結果をキャッシュし、キャッシュされていれば該当メソッドが処理されず、結果のみを返します。

```
@Cacheable("getItems")↓  
public List<Item> getAllItems(){
```

- ▶ 商品コード別等のキー値別にキャッシュ管理する場合には、key属性を指定します。

```
@Cacheable(value="getItem", key="#p0")↓  
public Optional<Item> getItem(Long itemId) {
```

キー値の指定には、以下のいずれかで指定する必要があります。

#p0

#a0

#itemId

※指定を誤ると、  
java.lang.IllegalArgumentExceptionが  
発生します。

# キャッシュを削除する

「@CacheEvict」でキャッシュを削除します。複数のキャッシュ削除を指定したい場合、「@Cache」を利用し、evict属性にて複数指定します。

- ▶ @CacheEvictでキャッシュの値を削除します。以下の例は、新規登録時に、全件検索取得のキャッシュを削除しています。

```
@CacheEvict(value="getItem", allEntries=true)  
public void addItem (Item item) {
```

allEntries = true  
指定した文字列で管理されているキャッシュ全体を削除したい場合に指定

- ▶ 複数のキャッシュ削除を指定したい場合は、@Cachingアノテーションにevict属性を用いて@CacheEvictを複数指定することができます。以下指定は、指定したデータの更新時に、対象のキー値で管理されているキャッシュがあれば削除、全件検索取得のキャッシュをそれぞれ削除しています。

```
@Caching(evict = {  
    @CacheEvict(value="getItem", key="#p0"),  
    @CacheEvict(value="getItem", allEntries=true)  
})  
public void updateItem(Long itemId, Item item) {
```



# 実機演習の流れ

- 1) spring3itemプロジェクトのpom.xmlにキャッシュを利用するための依存関係を追加
- 2) Spring3itemApplicationに「@EnableCaching」アノテーションを付与し、キャッシュを有効にする
- 3) ItemServiceクラスの商品検索メソッドに「@Cacheable」アノテーションでキャッシュを設定する
- 4) 動作確認を行い、キャッシュの効果を確認します
- 5) ItemServiceクラスの新規登録メソッドに「@CacheEvict」アノテーションでキャッシュを削除する
- 6) ItemServiceクラスの更新・削除登録メソッドに「@Caching」「@CacheEvict」アノテーションで複数のキャッシュを削除する
- 7) 動作確認を行い、適切なタイミングでキャッシュが削除されている事を確認します