

데이터 입력

Helper 메소드

- SeedStarter 엔티티

```
public class SeedStarter {  
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)  
    @Column(name = "SEED_STARTER_ID")  
    private Long id;  
  
    ..  
    ..  
  
    public void setFeature(Feature feature){  
        this.features.add(feature);  
        feature.setSeedStarter(this);  
    }  
    public void setDetail(Detail detail){  
        this.details.add(detail);  
        detail.setSeedStarter(this);  
    }  
}
```

Helper 메소드

- Feature 엔티티

```
public void setSeedStarter(SeedStarter seedStarter){  
    this.seedStarter = seedStarter;  
    seedStarter.getFeatures().add(this);  
}
```

- Detail 엔티티

```
public void setSeedStarter(SeedStarter seedStarter){  
    this.seedStarter = seedStarter;  
    seedStarter.getDetails().add(this);  
}
```

SeedStarter 전체

```
public class SeedStarter {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "SEED_STARTER_ID")
    private Long id;

    private LocalDateTime datePlanted;
    private boolean covered;
    @Enumerated(EnumType.STRING)
    private Type type;

    @OneToMany(mappedBy = "seedStarter", cascade = CascadeType.PERSIST, orphanRemoval = true)
    private List<Feature> features = new ArrayList<>();

    @OneToMany(mappedBy = "seedStarter", cascade = CascadeType.PERSIST, orphanRemoval = true)
    private List<Detail> details = new ArrayList<>();

    public void setFeature(Feature feature){
        this.features.add(feature);
        feature.setSeedStarter(this);
    }
    public void setDetail(Detail detail){
        this.details.add(detail);
        detail.setSeedStarter(this);
    }
}
```

```
public enum Type {
    PLASTIC, WOOD
}
```

Feature 전체

```
public class Feature {  
    @Id  
    @GeneratedValue(strategy = GenerationType.IDENTITY)  
    @Column(name = "FEATURE_ID")  
    private Long id;  
    @Enumerated(EnumType.STRING)  
    private FeatureType name;  
  
    @ManyToOne(fetch = FetchType.LAZY)  
    @JoinColumn(name = "SEED_STARTER_ID")  
    private SeedStarter seedStarter = new SeedStarter();  
  
    public void setSeedStarter(SeedStarter seedStarter){  
        this.seedStarter = seedStarter;  
        seedStarter.getFeatures().add(this);  
    }  
}
```

```
public enum FeatureType {  
    SUBSTRATE, FERTILIZER, PH_CORRECTOR  
}
```

Detail 전체

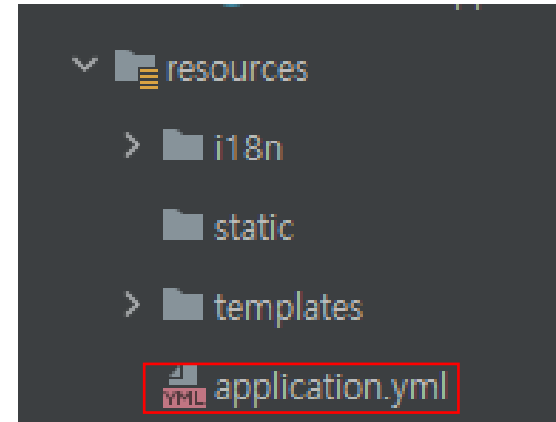
```
public class Detail {  
    @Id  
    @GeneratedValue(strategy = GenerationType.IDENTITY)  
    @Column(name = "DETAIL_ID")  
    private Long id;  
    private int rowNum;  
    private int seedPerCell;  
    private String variety;  
    @ManyToOne(fetch = FetchType.LAZY)  
    @JoinColumn(name = "SEED_STARTER_ID")  
    private SeedStarter seedStarter;  
  
    public void setSeedStarter(SeedStarter seedStarter){  
        this.seedStarter = seedStarter;  
        seedStarter.getDetails().add(this);  
    }  
}
```

데이터베이스 준비

데이터베이스 준비

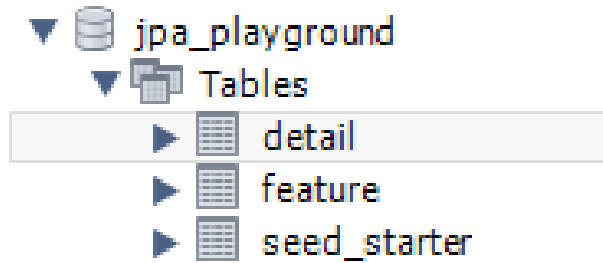
- application.yml

```
spring:
  datasource:
    driver-class-name: com.mysql.cj.jdbc.Driver
    url: jdbc:mysql://localhost:3306/jpa_playground?serverTimezone=Asia/Seoul
    username: kim
    password: sungryulKim12
  jpa:
    database-platform: org.hibernate.dialect.MySQL8Dialect
    hibernate:
      ddl-auto: create
    show-sql: true
    properties:
      hibernate.format_sql: true
```



데이터베이스 준비

- 스프링 부트 어플리케이션을 실행하여 테이블 생성 확인



	seed_starter_id	covered	date_planted	type
•	NULL	NULL	NULL	NULL

SEED_STARTER

	feature_id	name	seed_starter_id
•	NULL	NULL	NULL

FEATURE

	detail_id	row_num	seed_per_cell	variety	seed_starter_id
•	NULL	NULL	NULL	NULL	NULL

DETAIL

데이터베이스 준비

- 테스트 데이터 입력

- SEED_STARTER테이블에 데이터 입력

```
INSERT INTO SEED_STARTER(covered,date_planted,type) VALUES(0,'2002-04-05 10:10:10','PLASTIC');  
INSERT INTO SEED_STARTER(covered,date_planted,type) VALUES(0,'2001-04-05 10:10:10','WOOD');  
INSERT INTO SEED_STARTER(covered,date_planted,type) VALUES(1,'2003-04-05 10:10:10', 'ALL');
```

- 테이블 확인

	seed_starter_id	covered	date_planted	type
▶	1	0	2002-04-05 10:10:10.000000	PLASTIC
	2	0	2001-04-05 10:10:10.000000	WOOD
	3	1	2003-04-05 10:10:10.000000	ALL

데이터베이스 준비

- 테스트 데이터 입력

- FEATURE테이블에 데이터 입력

```
INSERT INTO FEATURE(name,SEED_STARTER_ID) VALUES('Seed starter-specific substrate',1);  
INSERT INTO FEATURE(name,SEED_STARTER_ID) VALUES('PH Corrector used',1);  
INSERT INTO FEATURE(name,SEED_STARTER_ID) VALUES('Fertilizer used',2);  
INSERT INTO FEATURE(name,SEED_STARTER_ID) VALUES('PH Corrector used',3);
```

- 테이블 확인

	feature_id	name	seed_starter_id
▶	1	Seed starter-specific substrate	1
	2	PH Corrector used	1
	3	Fertilizer used	2
	4	PH Corrector used	3

데이터베이스 준비

- 테스트 데이터 입력

- SEED_STARTER와 Feature의 정합성 확인

```
SELECT *  
FROM SEED_STARTER AS s  
NATURAL JOIN FEATURE AS f;
```

	seed_starter_id	covered	date_planted	type	feature_id	name
▶	1	0	2002-04-05 10:10:10.000000	PLASTIC	1	Seed starter-specific substrate
	1	0	2002-04-05 10:10:10.000000	PLASTIC	2	PH Corrector used
	2	0	2001-04-05 10:10:10.000000	WOOD	3	Fertilizer used
	3	1	2003-04-05 10:10:10.000000	ALL	4	PH Corrector used

데이터베이스 준비

- 테스트 데이터 입력

- Detail 데이터 입력

```
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(1,10,'Thymus vularis',1);
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(2,15,'Thymus pseudolaginosus',1);
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(3,20,'x citriodorus',1);
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(1,5,'Thymus herba-barona',2);
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(2,5,'Thymus serpyllum',2);
INSERT INTO DETAIL(ROW_NUM,SEED_PER_CELL, VARIETY, SEED_STARTER_ID) VALUES(1,20,'New Variety',3);
```

	detail_id	row_num	seed_per_cell	variety	seed_starter_id
▶	1	1	10	Thymus vularis	1
	2	2	15	Thymus pseudolaginosus	1
	3	3	20	x citriodorus	1
	4	1	5	Thymus herba-barona	2
	5	2	5	Thymus serpyllum	2
	6	1	20	New Variety	3

데이터베이스 준비

- 테스트 데이터 입력

- SEED_STARTER와 DETAIL 정합성 확인

```
SELECT *  
FROM SEED_STARTER AS s  
NATURAL JOIN DETAIL AS d;
```

	seed_starter_id	covered	date_planted	type	detail_id	row_num	seed_per_cell	variety
▶	1	0	2002-04-05 10:10:10.000000	PLASTIC	1	1	10	Thymus vulgaris
	1	0	2002-04-05 10:10:10.000000	PLASTIC	2	2	15	Thymus pseudolaginosus
	1	0	2002-04-05 10:10:10.000000	PLASTIC	3	3	20	x citriodorus
	2	0	2001-04-05 10:10:10.000000	WOOD	4	1	5	Thymus herba-barona
	2	0	2001-04-05 10:10:10.000000	WOOD	5	2	5	Thymus serpyllum
	3	1	2003-04-05 10:10:10.000000	ALL	6	1	20	New Variety

데이터베이스 준비

• 테스트 데이터 입력

▪ 전체 데이터 정합성 확인

```
SELECT *
FROM SEED_STARTER AS s
NATURAL JOIN FEATURE AS f
NATURAL JOIN DETAIL AS d;
```

Date planted	Covered	Type	Features	Rows		
03/20/2011	yes	Wood	Seed starter-specific substrate, PH Corrector used	1	Thymus vulgaris	10
				2	Thymus pseudolaginosus	15
				3	Thymus x citriodorus	20
03/25/2011	no	Plastic	Fertilizer used	1	Thymus herba-barona	5
				2	Thymus serpyllum	10

seed_starter_id	covered	date_planted	type	feature_id	name	detail_id	row_num	seed_per_cell	variety
1	0	2002-04-05 10:10:10.000000	PLASTIC	1	Seed starter-specific substrate	1	1	10	Thymus vularis
1	0	2002-04-05 10:10:10.000000	PLASTIC	1	Seed starter-specific substrate	2	2	15	Thymus pseudolagii
1	0	2002-04-05 10:10:10.000000	PLASTIC	1	Seed starter-specific substrate	3	3	20	x citriodorus
1	0	2002-04-05 10:10:10.000000	PLASTIC	2	PH Corrector used	1	1	10	Thymus vularis
1	0	2002-04-05 10:10:10.000000	PLASTIC	2	PH Corrector used	2	2	15	Thymus pseudolagii
1	0	2002-04-05 10:10:10.000000	PLASTIC	2	PH Corrector used	3	3	20	x citriodorus
2	0	2001-04-05 10:10:10.000000	WOOD	3	Fertilizer used	4	1	5	Thymus herba-baro
2	0	2001-04-05 10:10:10.000000	WOOD	3	Fertilizer used	5	2	5	Thymus serpyllum
3	1	2003-04-05 10:10:10.000000	ALL	4	PH Corrector used	6	1	20	New Variety

컨트롤러 준비

컨트롤러 준비

- 컨트롤러 코드

```
@Controller
public class SeedStarterMngController {

    @Autowired
    private SeedStarterService seedStarterService;

    @RequestMapping("/{"/seedstartermng"})
    public String showSeedstarters(final SeedStarter seedStarter, Model model) {
        List<SeedStarter> all = seedStarterService.findAll();
        model.addAttribute("allSeedStarters", all);
        return "seedstartermng";
    }
}
```

```
@Controller
@RequiredArgsConstructor
public class SeedStarterMngController {

    private final SeedStarterService seedStarterService;
```