

操作手册 — 测试流程与实验步骤

本手册给出 7 项功能的系统化测试流程：每项包含 Admin 前置操作、测试步骤（包括在 Admin 或 SQL 中执行的命令示例）、预期效果与检查点，便于手动验证系统行为与数据库规则。

测试项 1：自定义完整性约束条件的验证（示例：性别/年龄）

目的：验证模型/数据库对字段的约束能在录入时阻止非法数据。

前置（Admin）：

- 在 Django Admin 中登录管理员账号，打开 Users 或自定义测试模型的新增界面。

测试步骤：

- 在 Admin 中尝试创建一条记录，将 性别 字段填为非法值（例如 'UnknownX'，模型只允许 'M'/'F'）。点击保存。
- 在 Admin 中尝试创建 年龄=-5 或 年龄=1000 （若模型限制范围为 0-120）。点击保存。
- 在数据库控制台（例如 MySQL）执行插入 SQL 尝试绕过应用层约束，观察约束是否在 DB 层触发（如果在 DB 定义了 CHECK 约束）。

示例 SQL：

```
-- 尝试直接插入错误年龄（如果表有 CHECK 约束将失败）  
INSERT INTO Users (Username, Email, Full_Name, Hashed_Password, Is_Active) VALUES ('testx', 't@
```

预期效果与检查点：

- 在 Admin 保存时，若模型字段定义了 validators，应出现表单校验错误并阻止保存。
- 如果数据库定义了 CHECK 约束或触发器，直接在 DB 插入非法数据应被拒绝（出现 SQL 错误）。
- 检查点：Admin 页面显示错误信息；DB 插入命令返回错误或未写入记录。

测试项 2：利用通配符的模糊搜索

目的：验证 aggregate_search 或 Admin 的搜索能正确匹配模糊查询。

前置 (Admin):

- 通过 Admin 插入若干测试 ContainerMaster、Booking、Party 记录（包含易被模糊匹配的关键字，例如 "ABC123"、"上海港"、"maersk"）。

测试步骤:

1. 在系统的聚合搜索页面（ /admin/search/ 或自定义聚合搜索）输入关键词 ABC 或 sha ，点击搜索。
2. 在 Admin 的列表页面使用搜索框，输入部分字符串并确认返回结果包含预期记录。

示例：在 aggregate_search 中手动调用：

```
# 视图内部等价查询片段
```

```
ContainerMaster.objects.filter(Q(container_number__icontains='ABC') | Q(owner_party_id__party_n
```

预期效果与检查点:

- 搜索结果应包含匹配到的记录（部分匹配），并且没有抛出异常。
- 检查点：聚合搜索页面返回的 containers/bookings/tasks 列表包含预期项。

测试项 3：通过索引提高查询速度

目的：验证对关键列（如 Container_Master.Container_Number , Yard_Slot.Slot_Coordinates ）建索引后查询性能改善。

前置 (DB):

- 确认表上已有索引（如 UQ_Container_Number , UQ_Slot_Coordinates ）。可使用 `SHOW INDEX FROM Container_Master;` 查看。

测试步骤:

1. 在 DB 中运行没有索引的等价查询（如果可能，临时删除索引并测试），记录执行时间： `EXPLAIN ANALYZE SELECT ...` 。
2. 重新创建索引或使用现有索引，然后再次运行相同查询，比较耗时差异。

示例 SQL:

```
EXPLAIN ANALYZE SELECT * FROM Container_Master WHERE Container_Number LIKE 'ABCD%';
```

预期效果与检查点：

- 使用索引时 EXPLAIN 显示 index scan/seek 而非 full table scan，耗时显著降低。
- 检查点：EXPLAIN ANALYZE 输出中看到 Using index 或成本下降。

测试项 4：一定条件下触发器的使用

目的：测试数据库触发器在特定 DML（INSERT/UPDATE/DELETE）下是否触发并执行预期逻辑。

前置（DB）：

- 确认 sqlutil/triggers.sql 中有关触发器已被应用到数据库（或查看 SHOW TRIGGERS）。

测试步骤（示例触发器：当 Task 状态更新为 Completed 时更新 YardSlot）：

1. 在 Admin 中创建或找到一个 Task，其 status 为 InProgress 且 to_slot_id 指向某个 YardSlot。
2. 在 Admin 中将该 Task 的 status 修改为 Completed 并保存。
3. 在数据库或 Admin 中检查对应 YardSlot.current_container_id 是否被清空或 slot_status 是否变为 Available（触发器预期行为）。

示例 SQL（验证触发器执行）：

```
SELECT Slot_Status, Current_Container_ID FROM Yard_Slot WHERE Slot_ID = <to_slot_id>;
```

预期效果与检查点：

- 在修改任务状态后，触发器应自动执行并更新相关表。
- 检查点：Yard_Slot 中状态或 container 引用已按触发器逻辑变更；无需额外手动操作。

测试项 5：外连接、统计查询等复杂查询

目的：测试系统对复杂查询（LEFT JOIN、GROUP BY、聚合函数）的支持与性能。

前置（DB）：

- 在 Admin 中确保 Task、Container_Master、Yard_Slot 等表有样本数据。

测试步骤：

1. 在 DB 中直接运行统计 SQL，例如统计每个堆场区的已占用与空闲箱位数：

```
SELECT yb.Block_Name,  
       COUNT(slot.Slot_ID) AS total_slots,  
       SUM(CASE WHEN slot.Current_Container_ID IS NOT NULL THEN 1 ELSE 0 END) AS occupied  
FROM Yard_Block yb  
LEFT JOIN Yard_Stack ys ON ys.Block_ID = yb.Block_ID  
LEFT JOIN Yard_Slot slot ON slot.Stack_ID = ys.Stack_ID  
GROUP BY yb.Block_ID, yb.Block_Name;
```

2. 在 Admin 或自定义报表中比较前端渲染结果与 SQL 输出是否一致。

预期效果与检查点：

- SQL 能正确返回统计数据，且与前端展示一致。
- 检查点：GROUP BY 结果数值校验，通过样本比对确认正确性。

测试项 6：安全性的视图与用户权限运用

目的：验证基于 DB 视图与业务权限控制的访问限制有效性。

前置（Admin/DB）：

- 准备两个测试账号：viewer（仅查看相关视图）与 operator（具有操作权限）。
- 确认 View_User_Permissions、View_Yard_Inventory_Live 等视图存在。

测试步骤：

1. 使用 viewer 登录 Admin，尝试访问管理 Users 或 Permissions 页面，应被禁止或不可见（UI 隐藏）。
2. 使用 viewer 访问仪表盘或堆场视图，确认只读数据可见但无编辑入口。
3. 使用 operator 登录并尝试创建或编辑任务，验证权限允许操作。

预期效果与检查点：

- 权限不足用户无法访问受限页面或操作；视图/报表按权限过滤显示。
- 检查点：Admin 菜单与页面操作权限、数据库 User_Permissions 表、视图 View_User_Permissions 返回的权限列表。

测试项 7：基本表之间的联动（触发器/级联更新等）

目的：验证对某表的操作会否按照设计联动更新其他表（例如删除 container 后释放箱位）。

前置 (Admin/DB):

- 找到一个 ContainerMaster 已放置在某 YardSlot (`slot.current_container_id` 指向该 container)。

测试步骤:

1. 在 Admin 中删除该 ContainerMaster (或在 DB 中执行 DELETE)。
2. 检查对应 YardSlot.current_container_id 是否被设置为 NULL 或 slot_status 更新为 Available (取决于外键约束与触发器的业务逻辑)。
3. 若采用级联删除 (CASCADE), 验证相关日志与审计记录 (若有) 记录此操作。

预期效果与检查点:

- 删除 container 后, 对应箱位状态应变为可用或清空引用; 若设置了触发器或级联规则, 则会自动完成。
- 检查点: YardSlot 的 Current_Container_ID、Slot_Status、以及触发器日志或审计表的条目。

备注: 以上步骤在不同环境 (本地 dev / staging / production) 执行前应先在备份数据库或测试库验证, 避免误操作导致数据丢失。需要时我可以把这些测试步骤逐步转换为可执行的脚本或 Makefile 任务, 便于自动化测试运行。你要我现在把这些步骤写入仓库的 doc/操作手册.md (已创建) 并提交, 还是先调整内容?