

## **DATA 604 Assignment – 2**

### **Maruthi Mutnuri**

#### **Assumptions for Task 1:**

- PackageID is created for the unique combination of package name and package version
- One license can be used by multiple packages
- Python Package can have one or many authors, but authors can contribute to none or many python packages
- License can have one or many authors, but authors can contribute to none or many Licenses

#### **Assumptions for Task 2:**

- Relationship between Python Package table and License table is non-identifying because there is no FK in the PK.

#### **Task 3:**

##### **Python Package table:**

Constraints:

1. Package ID is unique for the Package Name and Package Version combination and cannot have null values. (entity integrity)
2. License ID is a mandatory field (Business Rule, stated in the question) and it should have a matching primary key in the License table. (referential integrity)

Example of task:

1. Duplicating the Package ID would violate the first constraint.
2. License ID value in the Python Package table does not have a matching value in the License table (orphan record) would violate the second constraint.

##### **License table:**

Constraints:

1. License ID is unique and cannot have null values. (entity integrity)
2. Short Name is a mandatory field and cannot have null values. (Business Rule)

Example of task:

1. Duplicating the License ID would violate the first constraint.
2. Leaving the Short Name blank would violate the second constraint.

##### **Author table:**

Constraints:

1. Username is unique and cannot have null values. (entity integrity)
2. Both First name and Last name cannot have null values. (Business Rule)

Example of task:

1. Duplicating the Username would violate the first constraint
2. Leaving the both First name and Last name blank would violate the second constraint

**Python Package Author table:**

Constraints:

1. Username should have a matching primary key in the Author table. (referential integrity)
2. Package ID should have a matching primary key in the Python Package table. (referential integrity)

Example of task:

1. Username value in the Python Package Author table does not have a matching value in the Author table (orphan record) would violate the first constraint.
2. Package ID value in the Python Package Author table does not have a matching value in the Python Package table (orphan record) would violate the second constraint.

**License Author table:**

Constraints:

1. Username should have a matching primary key in the Author table. (referential integrity)
2. License ID should have a matching primary key in the License table. (referential integrity)

Example of task:

1. Username value in the License Author table does not have a matching value in the Author table (orphan record) would violate the first constraint.
2. License ID value in the License Author table does not have a matching value in the License table (orphan record) would violate the second constraint.

**Requirements table:**

Constraints:

1. Package ID should have a matching primary key in the Python Package table. (referential integrity)
2. Referenced Package ID should have a matching primary key in the Python Package table. (referential integrity)

Example of task:

1. Package ID value in the Requirements table does not have a matching value in the Python Package table (orphan record) would violate the first constraint.
2. Referenced Package ID value in the Requirements table does not have a matching value in the Python Package table (orphan record) would violate the second constraint.

**Task 4: Normal Form of tables:**

1. **Python Package table: 2NF**, because there is a nested relation
2. **License table: 1NF**, because there are no multi valued attributes or no nested relations
3. **Author table: 1NF**, because there are no multi valued attributes or no nested relations
4. **Python Package Author table: 1NF**, because there are no multi valued attributes or no nested relations
5. **License Author table: 1NF**, because there are no multi valued attributes or no nested relations
6. **Requirements table: 1NF**, because there are no multi valued attributes or no nested relations

Because all the tables are either 1NF or 2NF no normalization is required.