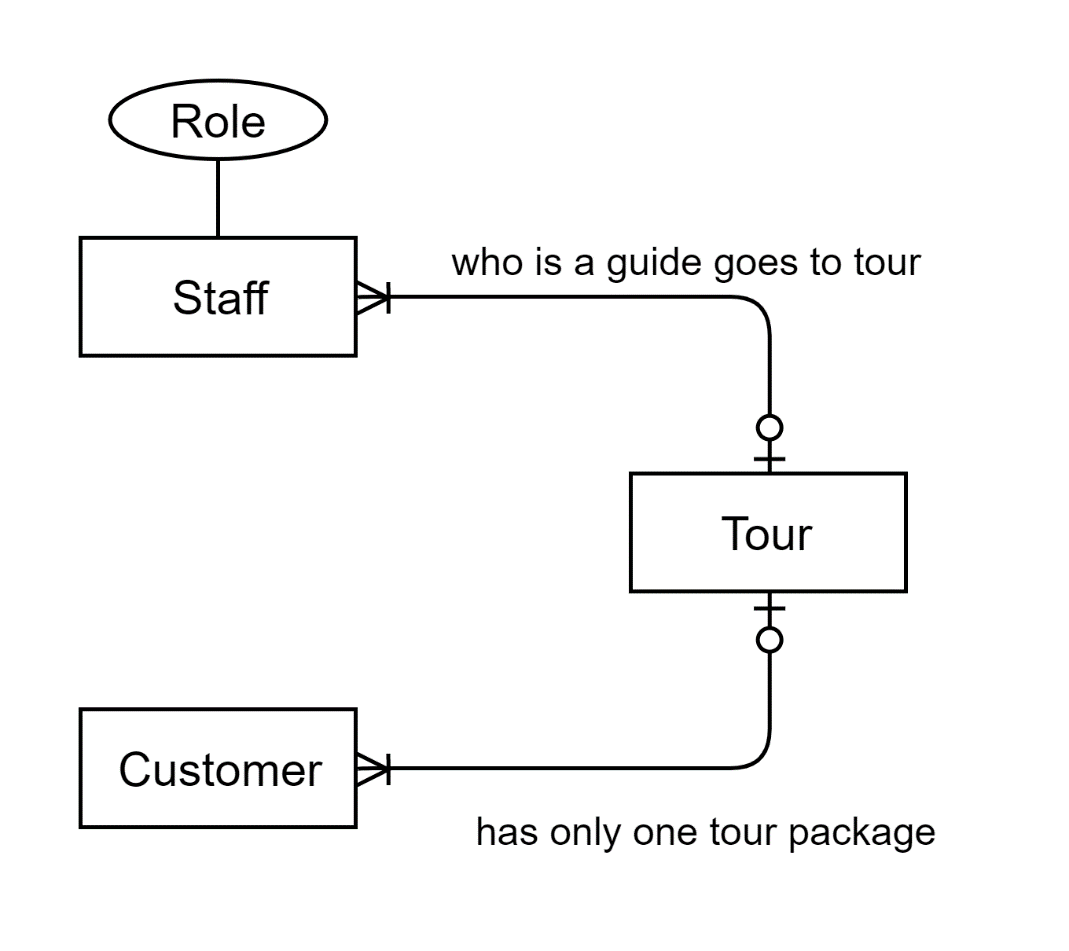
**Textual Analysis**

According to the scenario the tour company have the following relation between the entities:

1. Staffs have different roles assigned one for each
2. Multiple Staffs (specifically tour guides) could be assigned for a tour
3. Customer can only take one package from multiple available packages.
4. The tour and package are taken as a same entity
5. A registered customer might or might not have a tour assigned
6. Role is taken as an attribute of staff



**UNF**

Storing the data given in unique rows in a table we can observe:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Package ID** | **Package Name** | **Destination** | **Total No of Days** | **Difficulty** |
| LK25A | ABC | Annapurna Base Camp | 7 | Moderate |
| UI32A | Ghandruk | Ghandruk | 4 | Moderate |
| UI32A | Ghandruk | Pokhara | 4 | Moderate |
| NB34G | Everest Short Trek | Khumjung | 4 | Hard |
| NB34G | Everest Short Trek | Lukla | 4 | Hard |

From the above table we can observe that the destination column has repeating groups in it:

UNF: (**Package\_ID**, Pakage\_Name, {Destination}, Total\_Days, Difficulty)

Destination is the repeating group because a package might have multiple destinations for example Pokhara and Ghandruk. Package\_ID is taken as the primary key as it can uniquely identify all columns.

**1NF**

As Pakage\_Name, Total\_Days, and Difficulty depends upon the Package\_ID, it is assigned as a primary key.

Package\_ID 🡺 Pakage\_Name, Total\_Days, Difficulty

Removing the repeating group and creating the Destination Package and assigning package\_ID and Destination\_ID as composite key and creating the first normal form.

Package (**Package\_ID**, Pakage\_Name, Total\_Days, Difficulty)

Destination\_Package (**Package\_ID\*, Destination**)

**2NF**

There are no Partial dependencies as all of the non key elements are directly dependent on their respective primary key

Package\_ID 🡺 Pakage\_Name, Total\_Days, Difficulty

Package\_ID, Destination gives the unique value for every package-destination pair

Package (**Package\_ID**, Pakage\_Name, Total\_Days, Difficulty)

Destination\_Package (**Package\_ID\***, **Destination**)

**3NF**

There are no transitive dependencies as every none key entries are dependent on the primary key assigned for each table.

Package (**Package\_ID**, Pakage\_Name, Total\_Days, Difficulty)

Destination\_Package (**Package\_ID\***, **Destination**)

Since there are no transative dependencies this is the final normalized form

**No**

Adding Package ID, Activity ID and storing the data given in unique rows in a table we can observe:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Package ID** | **Package Name** | **Start Date** | **End Date** | **Guide** | **Day** | **Travel Details** | **Travel Mode** | **Dfficulty** | **Activity ID** | **Activities** | **Status** |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 1 | Kathmandu to Pokhara | Bus | Easy | A1 | Driving from KTM to Pokhara. | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 1 | Kathmandu to Pokhara | Stay | Easy | A2 | Overnight stay in Hotel. | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 2 | Pokhara to Ghandruk | Walk | Hard | A3 | Trek to Ghandruk. | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 2 | Pokhara to Ghandruk | Walk | Hard | A4 | Explore the Ghandruk Village. | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 3 | Ghandruk to Pokhara | Walk | Easy | A5 | View the sunrise and Himalayas. | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 3 | Ghandruk to Pokhara | Walk | Easy | A6 | Trek Down to Pokhara | Complete |
| UI32A | Ghandruk | 1st Jan 2019 | 7th Jan 2019 | Will Stark | Day 4 | Pokhara to Kathmandu | Bus | Moderate | A7 | Drive back to Kathmandu | Remaining |

From the above table we can observe that the Day, Travel Details, Activity ID, Activities, Status, travel Mode and Difficuly are the repeating groups and Activity\_ID, Activities, Status and Travel mode are also repeating groups.

UNF: (**Package\_ID**, Package\_Name, Start\_Date, End\_Date, Tour\_Guide, {Day, Travel\_Details, {Activity\_ID, Activities, Status, Travel\_Mode, Difficulty\_Level}})

Now separating the Repeating group to a new table in 1NF.

**1NF**

Package ID is taken as the Primary key as it gives

Package\_ID 🡺 Package\_Name, Start\_Date, End\_Date, Tour\_Guide

Package\_ID, Day 🡺 Travel\_Details, Activity\_ID

Package\_ID, Day, Activity\_ID 🡺 Activities, Status, Travel\_Mode, Difficulty\_Level

Removing the repeating group and creating the Itenary\_Tour. Assigning package\_ID as Primary key and creating the first normal form.

Tour (**Package\_ID**, Package\_Name, Start\_Date, End\_Date, Tour\_Guide)

Itenary\_Tour (**Package\_ID**\*, **Day**,Travel\_Details, Activity\_ID)

Day\_Activities (**Package\_ID**\*, **Day\***, **Activity\_ID**, Activities, Status, Travel\_Mode, Difficulty\_Level)

**2NF**

The only partial dependency encountred here is in Day Activities, where activity ID gives all the other attributes

Package\_ID 🡺 Package\_Name, Start\_Date, End\_Date, Tour\_Guide

Package\_ID, Day 🡺 Travel\_Details, Activity\_ID

Package\_ID, Day, Activity\_ID 🡺 Activities, Status, Travel\_Mode, Difficulty\_Level

Activity\_ID 🡺 Activities, Status, Travel\_Mode, Difficulty\_Level

Tour (**Package\_ID**, Package\_Name, Start\_Date, End\_Date, Tour\_Guide)

Itenary\_Tour (**Package\_ID**\*, **Day**,Travel\_Details, Activity\_ID)

Day\_Activities (**Package\_ID**\*, **Day\***, **Activity\_ID\***)

Activities(**Activity\_ID**, Activities, Status, Travel\_Mode, Difficulty\_Level)

**3NF**

There are no transitive dependencies as every none key entries are dependent on the primary key assigned for each table.

Tour (**Package\_ID**, Package\_Name, Start\_Date, End\_Date, Tour\_Guide)

Itenary\_Tour (**Package\_ID**\*, **Day**,Travel\_Details, Activity\_ID)

Day\_Activities (**Package\_ID**\*, **Day\***, **Activity\_ID**, Activities, Status, Travel\_Mode, Difficulty\_Level

Since there are no transative dependencies this is the final normalized form