**Introduction**

The BT database aims to be a flexible solution for a telecommunications company. The database has multiple levels of authentication depending on the role of the employee and features an interface designed for simplicity.

**Table Analysis**

The BT database requires tables to store records within; this database is concerned with employees, customers, packages, employee roles and package discounts.

**Employee Table**

The Employees table would need the following information:

Employee ID (Integer) (Primary Key) (Auto Number)  
First Name (Text)  
Last Name (Text)  
Password (Text)  
Email Address (Text)  
Address One (Text)  
Address Two (Text)  
City (Text)  
Post Code (Text)  
Phone Number (Text)  
Gender (Text)  
Role (Text)  
Hire Date (Short Date)

Considering that the gender and roles would be frequently repeating throughout records these are normalized into separate tables.  
  
Employee ID (Integer) (Primary Key) (Auto Number)  
First Name (Text)  
Last Name (Text)  
Password (Text)  
Email Address (Text)  
Address One (Text)  
Address Two (Text)  
City (Text)  
Post Code (Text)  
Phone Number (Text)  
*Gender (Integer) Now suitable for a foreign key.  
Role (Integer) Now Suitable for a foreign key.*  
Hire Date (Short Date)

**Customer Table**

The Customer table should contain the following information:

Customer ID (Integer) (Primary Key) (Auto Number)   
First Name (Text)   
Last Name (Text)  
Date of Birth (Short Date)  
Email Address (Text)  
Phone Number (Text)  
Mobile Number (Text)  
Address One (Text)  
Address Two (Text)  
City (Text)  
Post Code (Text)  
*Gender (Integer) Suitable for foreign key.*Package (Text)   
Contract Start (Short Date)   
Discount Code (Text)   
Discount Interval (Text)  
Account Number (Text)

Since the customers package, discount code, interval will be frequently occurring into the database these fields were normalized into a separate table within the database. Therefore the customer table was altered to this:

Customer ID (Integer) (Primary Key) (Auto Number)   
First Name (Text)   
Last Name (Text)  
Date of Birth (Short Date)  
Email Address (Text)  
Phone Number (Text)  
Mobile Number (Text)  
Address One (Text)  
Address Two (Text)  
City (Text)  
Post Code (Text)  
*Gender (Integer) Suitable for foreign key.*Package (Integer) Now suitable for foreign key.   
Contract Start (Short Date)   
*Discount Code (Integer) Now suitable for foreign key.  
Discount Interval (Text) Now Suitable for foreign key.*  
Account Number (Text)

**Packages Table**

The package table needs to contain information such as the name of the package and price. Also as these packages are for virtual products information about the speed and data limitations are also implemented.

Package ID (Integer) (Primary Key) (Auto Number)  
 Price (Currency)  
 Package Name (Text)  
 Bandwidth Limit (Integer)  
 Data Limit (Integer)  
 Dynamic IP (Boolean)

There is no sensible way to normalize this table, the Dynamic IP is likely to repeat but inserting this into a separate table would not enhance the structure of the database.

**Employee Roles**

The employee role table needs to link to the employees table so that each individual employee can be assigned a role. The employee roles need a name and the individual wages can be provided here, although, if the database is to be developed further this would not work well with pay rises.

The employee roles table structure is as follows:

Employee Role ID (Integer) (Primary Key) (Auto Number)   
Role Name (Text)   
Wages (Currency)

**Gender Table**

Discount Table

Occurrence Table

