Bank Management System
In Python

INTRODUCTION

ABOUT PYTHON:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting). Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible. Python 2 was discontinued with version 2.7.18 in 2020.

ABOUT MYSQL:

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds. Other kinds of data stores can also be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems. Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, Python, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table.
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

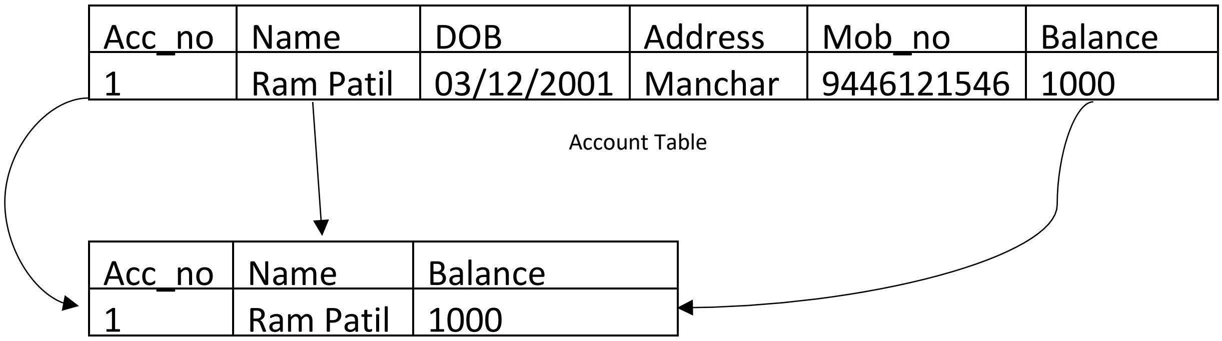
ABOUT RDBMS:

A relational database is a type of database that stores and provides access to data points that are related to one another. Relational databases are based on the relational model, an intuitive, straightforward way of representing data in tables. In a relational database, each row in the table is a record with a unique ID called the key. The columns of the table hold attributes of the data, and each record usually has a value for each attribute, making it easy to establish the relationships among data points.

We use Two Tables in this project

i. Account Table: This table contains Acc_no, Name, DOB, Address, Mob_no and Balance. We use Acc_no as **Primary Key** in this table and connect with other table by using **Foreign Key**.

ii. Amount Table: This table contains Acc_no, Name and Balance. We connect this table with Account table by using **Foreign Key**.



Acc_no	Name	DOB	Address	Mob_no	Balance
1	Ram Patil	03/12/2001	Manchar	9446121546	1000

Account Table

Acc_no	Name	Balance
1	Ram Patil	1000

Amount Table

ABOUT BANK MANAGEMENT SYSTEM PROJECT:

Bank Management System project is written in Python. This is a simple console based system which is very easy to understand and use. Talking about the system, it contains all the basic functions which include creating a new account, view account holders record, withdraws and deposit amount, balance inquiry and closing an account. In this mini project, there is no such login system. This means he/she can use all those available features easily without any restriction.

Talking about the features of the Bank Management System, a user can create an account by providing the name of the account holder, number and providing an initial opening amount. Then the user can also deposit and withdraw money just by providing his/her account and entering the amount. For certain purpose, he/she can also check for the balance inquiry which displays the account number and amount.

SOURCE CODE

MYSQL CODE:

```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.19 MySQL Community Server - GPL
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database Bank_sys ; Query
OK, 1 row affected (0.63 sec)

mysql> create table account(Acc_no int (10) primary key, Name varchar(255), Dob varchar(10),
Address varchar(255), Mob_no int(13), Balance int(10));
Query OK, 0 rows affected, 3 warnings (6.43 sec)

mysql> create table amount(Acc_no int(10), Name varchar(255), TotalBalance int(10), foreign
key(Acc_no) references account(Acc_no));
Query OK, 0 rows affected, 2 warnings (2.23 sec)

mysql> describe account;
+------------+-------------------+------+-------+----------+-------+
| Field      | Type              | Null | Key | Default | Extra |
+------------+-------------------+------+-------+----------+-------+
| Acc_no   | int               | NO  | PRI | NULL   |      |
| Name     | varchar(255) | YES |       | NULL   |      |
| Dob       | varchar(10)   | YES |       | NULL   |      |
| Address | varchar(255) | YES |      | NULL   |      |
| Mob_no| int               | YES |      | NULL   |      |
| Balance | int              | YES |      | NULL   |      |
+------------+-------------------+------+------+----------+-------+
6 rows in set (1.26 sec)

mysql> describe amount;
+------------------+-------------------+-------+--------+-----------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| Acc_no       | int          | YES  | MUL | NULL    |       |
| Name         | varchar(255) | YES  |     | NULL    |       |
| TotalBalance | int          | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+ 3
rows in set (0.01 sec)
```