# Business Description

## Business background

The Mountaineering club needs a database for storing their data and the information about the climbs.

## Problems. Current Situation

They need:

1. The date of the beginning and end of each climb
2. The names and addresses of the climbers involved,
3. The name and height of the mountain
4. The country, and the area where the mountain is located

That all must be recorded.

## The benefits of implementing a database. Project Vision

It’s nice to have any sort of database if you have a lot of clients. It can be also fun to build a model of one and implement it...

# Model description

## Definitions & Acronyms

## Logical Scheme



## Objects

Tables Description

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Climb** | ID | ID of each climb happening in the club, PK | Int |
| Start\_date | Start date of each climb | Date |
| End\_date | End date of each climb | Date |

Example:

|  |  |  |
| --- | --- | --- |
| ID | Start\_date | End\_date |
| 1 | 2021-03-24 | 2021-04-20 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Climbers** | Passport\_ID | ID of each climber, PK | Int |
| First\_name | First name of the climber | varchar |
| Last\_name | Last name of the climber | varchar |
| Address | Address of the climber | varhcar |

Example:

|  |  |  |  |
| --- | --- | --- | --- |
| Passport\_id | First\_name | Last\_name | Address |
| WR231209 | Andrzej | Goffman | Ul. Pilsudskiego, 29 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Mountains** | ID | Mountain ID, PK | Int |
| Mountain\_name | Name of the mountain | varchar |
| Mountain\_height | Height of the mountain | int |
| Location\_ID | ID of the location where the mountain is, FK | int |

Example:

|  |  |  |  |
| --- | --- | --- | --- |
| id | Mountain\_name | Mountain\_height | Location\_id |
| 2419 | Shkhara | 5102 | 543 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Route** | ID | Specific route ID, PK | Int |
| Route\_name | Name of the route | varchar |
| Route\_length | Length of the route | int |
| Mountain\_id | ID of the mountain where this route is, FK | int |

Example:

|  |  |  |  |
| --- | --- | --- | --- |
| id | Route\_name | Route\_length | Mountain\_id |
| 20 | Walk in skies | 10 | 2419 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Location** | ID | ID of specific location, PK | Int |
| Country | Name of the country where the mountain is located | varchar |
| Area/County | Exact area where the mountain is | varchar |

Example:

|  |  |  |
| --- | --- | --- |
| id | Country | Area/County |
| 4 | Georgia | Samegrelo-Zemo Svaneti |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Climbing\_climbers\_joint\_table** | Climb\_id | ID of each climb happening in the club, PK/FK | Int |
| Climber\_id | ID of each climber, PK\FK | int |

Example:

|  |  |
| --- | --- |
| Climb\_id | Climber\_id |
| 1 | WR231209 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Climbing\_mountains\_joint\_table** | Climb\_id | ID of each climb happening in the club, PK/FK | Int |
| Mountain\_id | ID of the mountain to climb, PK/FK | int |

Example:

|  |  |
| --- | --- |
| Climb\_id | Mountain\_id |
| 1 | 2419 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| **Climbing\_route\_joint\_table** | Climb\_id | ID of each climb happening in the club, PK/FK | Int |
| Route\_id | Specific route ID, PK/FK | int |

Example:

|  |  |
| --- | --- |
| Climb\_id | Route\_id |
| 1 | 20 |

Comments on table relationships:

This DB has 8 tables. 3 of them are junction tables, needed to implement many to many relationships.

**[Climb] to [Climbers] many to many:** because one climber can participate in many climbs and one climb can consist of several climbers. (It’s implemented using a joint table *- Climbing\_climbers\_joint\_table* **–** consisting of two foreign keys, combination of which is used as a the primary key)

**[Climb] to [Mountains] many to many:** because one climb can go through several mountains and on the one mountain can happen more than one climb. (It’s implemented using a joint table - *Climbing\_mountains\_joint\_table* **–** consisting of two foreign keys, combination of which is used as a the primary key)

**[Climb] to [Routes] many to many:** because one climb can go through several routes and the one route can be a part of several climbs. (It’s implemented using a joint table *- Climbing\_routes\_joint\_table –*consisting of two foreign keys, combination of which is used as a the primary key)

**[Mountains] to [Route] one to many:** because one mountain can have several routes on it, but not the other way around

**[Mountains] to [Location] one to many:** because one mountain can be in one exact location, but one location can include several mountains