# Table: Parts

|  |  |
| --- | --- |
| **Schema:** | SystemSchematicBuilder |
| **Responsible:** | Kurt A Vedros |

Table holds the part references to the individual parts table.

I.E. what the part is. All Foreign keys originally set to -1. What the part is, the foreign key is > 0.

This is so it’s a simple search of what is > 0 to get what the part is and its designated table.

Part – The individual component of the system that references a physical object.

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data type** | **Description** |
| PartID | Int | Primary Key to Parts Table. |
| SourceID | Int | Foreign Key to Sources Table. (If source is the part then this will be number > 0) |
| PipeID | Int | Foreign Key to Pipes Table. (If pipe is the part then this will be number > 0) |
| TankID | Int | Foreign Key to Tanks Table. (If tank is the part then this will be number > 0) |
| TubeID | Int | Foreign Key to Tubes Table. (If tube is the part then this will be number > 0) |
| ValveID | Int | Foreign Key to Valves Table. (If valve is the part then this will be number > 0) |
| PumpID | Int | Foreign Key to Pumps Table. (If pump is the part then this will be number > 0) |
| ExitID | Int | Foreign Key to Exits Table. (If exit is the part then this will be number > 0) |
| CordID | Int | Foreign Key to Cords Table. (If cord is the part then this will be number > 0) |
| SwitchBoxID | Int | Foreign Key to SwitchBoxes Table. (If switchbox is the part then this will be number > 0) |
| SwitchID | Int | Foreign Key to Switches Table. (If switch is the part then this will be number > 0) |
| SplitterID | Int | Foreign Key to Splitters Table. (If splitter is the part then this will be number > 0) |
| ConverterID | Int | Foreign Key to Converters Table. (If converter is the part then this will be number > 0) |

**Procedures (Database functions)**

* sprocGetParts
  + Returns a list of Part objects that are currently stored in the database.
* sprocGetPart
  + Returns a Part object that is currently stored in the database that has the given Part ID number.
* sproc\_PartAdd
  + Stores the given Part object into the database and returns its assigned Part ID number.
* sproc\_PartUpdate
  + Changes information of the Parts Table in the database with the given primary key to the given Part class object.
* sproc\_PartRemove
  + Removes the Part from the Parts Table with the given Part ID number.

**Additional Procedures (Database functions)**

* sprocGetAllPartIs
  + Returns the Value of the associated Part ID number.
* sprocGetValuesByPartID
  + Returns a list of all Parts that is of the given PartName. Note: PartName is by first letter cap the rest lower case, example Pipe, Tank or Cord.
* sprocGetConnentsOfPart
  + Returns all the Parts that are connected to the give Part ID number stored in PartToParts table.

**DAL (Software Functions)**

* PartsGetAll()
  + Returns a list of Part objects that are currently stored in the database.
* PartGet(int ID)
  + Returns a Part object that is currently stored in the database that has the given Part ID number.
* PartAdd(Part Part)
  + Stores the given Part object into the database and returns its assigned Part ID number. Returns -1 if error occurred, otherwise returns null.
* PartUpdate(Part Part)
  + Changes a Part object that is currently stored in the database with the given Part’s Part.ID with the Part’s information.
* PartRemove(int ID)
  + Removes the Part object stored in the database with the given Part ID number. Returns -1 if error occurred, otherwise returns null.

**Additional DAL (Software Functions)**

* GetValuesByPartID(int ID)
  + Returns the Value of the associated Part ID number.
* GetAllPartsIs(string PartName)
  + Returns a list of all Parts that is of the given PartName. Note: PartName is by first letter cap the rest lower case, example Pipe, Tank or Cord.
* GetConnectsOfPart(int ID)
  + Returns all the Parts that are connected to the give Part ID number stored in PartToParts table.