# Table: PartToParts

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

Table holds all information of the Parts connection to other Parts and its Tables.

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data type** | **Description** |
| PartToPartID | Int | Primary Key to Parts Table |
| FromPartID | Int | Primary Key to the Part that the connection is going from. |
| ToPartID | Int | Primary Key to the Part that the connection is going to. |

**Procedures (Database functions)**

* sprocGetPartToParts
  + Returns a list of PartToPart objects that are currently stored in the database.
* sprocGetPartToPart
  + Returns a PartToPart object that is currently stored in the database that has the given PartToPart ID number.
* sproc\_PartToPartAdd
  + Stores the given PartToPart object into the database and returns its assigned PartToPart ID number.
* sproc\_PartToPartUpdate
  + Changes information of the PartToParts Table in the database with the given primary key to the given PartToPart class object.
* sproc\_PartToPartRemove
  + Removes the PartToPart from the PartToParts Table with the given PartToPart ID number.

**DAL (Software Functions)**

* PartToPartsGetAll()
  + Returns a list of PartToPart objects that are currently stored in the database.
* PartToPartGet(int ID)
  + Returns a PartToPart object that is currently stored in the database that has the given PartToPart ID number.
* PartToPartAdd(PartToPart PartToPart)
  + Stores the given PartToPart object into the database and returns its assigned PartToPart ID number. Returns -1 if error occurred, otherwise returns null.
* PartToPartUpdate(PartToPart PartToPart)
  + Changes a PartToPart object that is currently stored in the database with the given PartToPart’s PartToPart.ID with the PartToPart’s information.
* PartToPartRemove(int ID)
  + Removes the PartToPart object stored in the database with the given PartToPart ID number. Returns -1 if error occurred, otherwise returns null.