* **Boundary Value Analysis (BVA)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Input** | **Classification** | **Expected Outcome** |
| Min Value | 0000000000 | Valid | Accept |
| Max Value | 9999999999 | Valid | Accept |
| Less than Min | -1 | Invalid | Reject, Error message indicating invalid number format |
| More than Max | 10000000000 | Invalid | Reject, Error message indicating invalid number format |
| Non-Numeric | abcdefghij | Invalid | Reject, Error message indicating invalid characters |

**Explanation of BVA:**

* We consider the minimum (0000000000) and maximum (9999999999) valid values based on the requirement of 10 digits.
* We test one value below the minimum (-1) and one value above the maximum (10000000000) to check for boundary issues.
* We include a non-numeric input (abcdefghij) to ensure the system rejects characters.
* **Equivalence Partitioning for the Scenario:**

1. **Valid Partition:**
   * Inputs in this partition are valid mobile numbers containing exactly 10 digits. For instance:
     + 1234567890
     + +1234567890
     + 0123456789
2. **Invalid Partition:**
   * Inputs in this partition are invalid because they don't meet the criteria of exactly 10 digits. This partition includes:
     + Less than 10 digits: 123456789
     + More than 10 digits: 12345678901
3. **Invalid Partition (Non-Numeric):**
   * This partition includes inputs that are not purely numeric. For example:
     + Alphanumeric input: abcdefghij
     + Special characters: !@#$%^&\*()

**EQUILANCE PARTITIONING**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Invalid** | **Valid** | **Invalid** |
| 1234567890 |  | 1234567890 |  |
| +1234567890 |  | +1234567890 |  |
| 0123456789 |  | 0123456789 |  |
| 123456789 | 123456789 |  |  |
| 12345678901 | 12345678901 |  |  |
| abcdefghij |  |  | abcdefghij |
| !@#$%^&\*() |  |  | !@#$%^&\*() |



