

# Test Cases

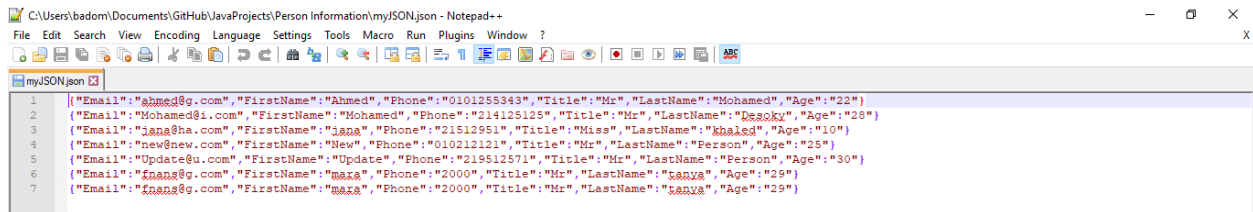
## TC#1:

**Title:** Importing a JSON file through system directory.

**Description:** the user can simply choose his JSON file through the browsing his system folders.

**Precondition:** the user should know where is the JSON file?

**Assumption:** the file should be stored as in key and value pairs.



```
1 [{"Email": "ahmed@g.com", "FirstName": "Ahmed", "Phone": "0101255343", "Title": "Mr", "LastName": "Mohamed", "Age": "22"}]
2 [{"Email": "Mohamed@i.com", "FirstName": "Mohamed", "Phone": "214125125", "Title": "Mr", "LastName": "Desoky", "Age": "28"}]
3 [{"Email": "jana@ha.com", "FirstName": "jana", "Phone": "21512951", "Title": "Miss", "LastName": "Khaled", "Age": "10"}]
4 [{"Email": "new@new.com", "FirstName": "New", "Phone": "010212121", "Title": "Mr", "LastName": "Person", "Age": "25"}]
5 [{"Email": "Update@u.com", "FirstName": "Update", "Phone": "219512571", "Title": "Mr", "LastName": "Person", "Age": "30"}]
6 [{"Email": "fana@g.com", "FirstName": "mara", "Phone": "2000", "Title": "Mr", "LastName": "tanya", "Age": "29"}]
7 [{"Email": "fana@g.com", "FirstName": "mara", "Phone": "2000", "Title": "Mr", "LastName": "tanya", "Age": "29"}]
```

## Test Steps:

1. Run the application
2. Click the 'Browse' button.
3. Choose your file from the directory.
4. Click the 'Import' Button.

**Expected Result:** A table showing all the records of your JSON file should appear in the right side of the application

Here's what this test case would look like in the application:

Person Information System

Person Information

First Name

Last Name

Title

Age

Phone

Email

Mr

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
Jana	Khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
mara	tanya	Mr	29	2000	fnans@g.com

orange™

Add

Update

Delete

Reset

Sort

First Name

Filter

Import JSON

C:/Users/badom/workspace/Person Information/myJSON.json

Browse

Exit

## TC#2:

**Title:** Importing a JSON file through copying and pasting the file path.

**Description:** the user can write down the path himself in the text field or if he knows it or already copied it he can paste it and go right away.

**Precondition:** the user should know where is the JSON file?

**Assumption:** the file should be stored as in key and value pairs.

```
1 [{"Email": "ahmed@g.com", "FirstName": "Ahmed", "Phone": "0101255343", "Title": "Mr", "LastName": "Mohamed", "Age": "22"}]
2 [{"Email": "Mohamed@i.com", "FirstName": "Mohamed", "Phone": "214125125", "Title": "Mr", "LastName": "Desoky", "Age": "28"}]
3 [{"Email": "jana@ha.com", "FirstName": "jana", "Phone": "21512951", "Title": "Miss", "LastName": "khaled", "Age": "10"}]
4 [{"Email": "new@new.com", "FirstName": "New", "Phone": "010212121", "Title": "Mr", "LastName": "Person", "Age": "25"}]
5 [{"Email": "Update@u.com", "FirstName": "Update", "Phone": "219512571", "Title": "Mr", "LastName": "Person", "Age": "30"}]
6 [{"Email": "fns@g.com", "FirstName": "mara", "Phone": "2000", "Title": "Mr", "LastName": "tanya", "Age": "29"}]
7 [{"Email": "fns@g.com", "FirstName": "mara", "Phone": "2000", "Title": "Mr", "LastName": "tanya", "Age": "29"}]
```

## Test Steps:

1. Run the application
2. Write down the path in the 'Text Field' next to the 'Import' button.
3. Click the 'Import' Button.

**Expected Result:** A table showing all the records of your JSON file should appear in the right side of the application.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c.
Mohamed	Desoky	Mr	28	214125125	Mohamed@i.
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c.
Update	Person	Mr	30	219512571	Update@u.c.
mara	tanya	Mr	29	2000	fns@g.com

### TC#3:

**Title:** Adding a new person information.

**Description:** the user can add any information about a person he wants.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** 'First Name', 'Last Name' should be alphabetical, 'Phone, Age' should be numeric and the email can be in any format.

The screenshot shows a web application titled "Person Information System". On the left, there is a form labeled "Person Information" with fields for First Name (Add), Last Name (Data), Title (Mr), Age (20), Phone (01012010210), and Email (add.data@g.com). Below the form are buttons for Add, Update, Delete, and Reset. On the right, there is a table with the following data:

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
mara	tanya	Mr	29	2000	fmans@g.com

An "orange" logo is visible in the top right corner of the interface.

### Test Steps:

1. Type in the data in 'First Name', 'Last Name', 'Age', 'Phone', 'Email' and choose the 'Title' from the combo box.
2. Click the 'Add' Button.

**Expected Result:** The table should show the new added row at the end of it and a message box pops saying that you have successfully added a row.

This screenshot shows the same interface as the previous one, but with a success message box displayed. The message box, titled "Person Information System", contains the text "Successfully added a new row!" and an "OK" button. The table now includes an additional row at the bottom:

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
mara	tanya	Mr	29	2000	fmans@g.com
Add	Data	Mr	20	01012010210	add.data@g...

Below the table, there are additional controls: a "Sort" button with a dropdown menu set to "First Name", a "Filter" button with an empty text box, and an "Import JSON" button with a text field containing the path "C:/Users/badom/workspace/Person Information/myJSON.json" and a "Browse" button. The "orange" logo remains in the top right corner.

#### TC#4:

**Title:** Update person information.

**Description:** the user can update any information about a person he wants.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** the user should select the row that he wants to update.

The screenshot shows the 'Person Information System' application. On the left, there is a form titled 'Person Information' with fields for First Name (New), Last Name (Time), Title (Mr), Age (50), Phone (01012125211), and Email (New.t@g.com). Below the form are buttons for Add, Update, Delete, and Reset. On the right, there is a table with columns: First Name, Last Name, Title, Age, Phone, and Email. The table contains several rows, with the row 'mala lanya Mr 29 2000 trans@g.com' highlighted. An 'orange' logo is visible in the top right corner.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
mala	lanya	Mr	29	2000	trans@g.com
Add	Data	Mr	20	01012010210	add.data@g...

#### Test Steps:

1. Select the Row you want to update.
2. Type in the data in 'First Name', 'Last Name', 'Age', 'Phone', 'Email' and choose the 'Title' from the combo box.
3. Click the 'Update' Button.

**Expected Result:** The table should show the new updated added row at the end of it and a message box pops saying that you have successfully updated a row.

This screenshot shows the same application as the previous one, but with a success message box displayed. The message box says 'Successfully Updated the row!' and has an 'OK' button. The table now includes a new row at the bottom: 'New Time Mr 50 01012125211 New.t@g.com'. The 'Add' button is now disabled, and the 'Update' button is highlighted. The 'orange' logo is still present in the top right corner.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...
New	Time	Mr	50	01012125211	New.t@g.com

## TC#5:

**Title:** Delete person information.

**Description:** the user can delete any record from the person information JSON file.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** the user should select the row that he wants to delete.

The screenshot shows the 'Person Information System' window. On the left, there is a form for adding or updating person information with fields for First Name, Last Name, Title (a dropdown menu currently showing 'Mr'), Age, Phone, and Email. On the right, there is a table displaying the current data. The 'Delete' button at the bottom is highlighted in blue. An 'orange' logo is visible in the top right corner.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...
New	Time	Mr	50	01012125211	New.k@g.com

## Test Steps:

1. Select the Row you want to delete.
2. Click the 'Delete' Button.

**Expected Result:** The table should show that selected row is deleted and a message box pops saying that you have successfully deleted the row.

This screenshot shows the same 'Person Information System' window, but with a modal dialog box in the center. The dialog box has a red 'X' icon and the text 'Successfully deleted the row!' with an 'OK' button. The table on the right now shows only five rows, as the row with 'New Time' has been removed. The 'Add' button at the bottom is highlighted in blue. An 'orange' logo is visible in the top right corner.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...

## TC#6:

**Title:** Sort person information by column headers.

**Description:** the user can sort the table based on any field he chooses.

**Precondition:** the user should have imported the JSON file already.

### Test Steps:

1. Click on the header of the column first time for ascending order.
2. Click on the header of the column second time for descending order.

The screenshot shows the 'Person Information System' window. On the left, there is a form for adding or updating person information with fields for First Name, Last Name, Title (a dropdown menu currently showing 'Mr'), Age, Phone, and Email. On the right, there is a table with the following data:

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khalel	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...

An 'orange' logo is visible in the top right corner of the window.

**Expected Result:** The table should sort all the rows based on the column you've clicked on its column's header.

The screenshot shows the 'Person Information System' window after sorting by First Name. The table data is now sorted alphabetically by First Name:

First Name	Last Name	Title	Age	Phone	Email
Add	Test	Ms	25	01222511654	add.test@t.c...
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
jana	khalel	Miss	10	21512951	jana@ha.com
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...

Below the table, there are several buttons: 'Add', 'Update', 'Delete', 'Reset', 'Sort', 'Filter', 'Import JSON', and 'Exit'. The 'Sort' button is active, and a dropdown menu next to it shows 'First Name'. The 'Filter' button has an empty text input field next to it. The 'Import JSON' button is next to a text field containing the file path 'C:/Users/badom/workspace/Person Information/myJSON.json', with a 'Browse' button next to it. The 'Exit' button is in the bottom right corner.

## TC#7:

**Title:** Sort person information by choosing a field.

**Description:** the user can sort the table bas on any field he chooses using the combo box in beside 'Sort' button.

**Precondition:** the user should have imported the JSON file already.

### Test Steps:

1. Choose the column from the combo box you want to sort.
2. Click on 'Sort' button.

Person Information System

Person Information

First Name:   
Last Name:   
Title:   
Age:   
Phone:   
Email:

Add Update Delete Reset

Sort Last Name

First Name	Last Name	Title	Age	Phone	Email
Add	Test	Ms	25	01222511654	add.test@t.c...
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
jana	khaled	Miss	10	21512951	jana@ha.com
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...

orange™

**Expected Result:** The table should sort all the rows based on the column you've chosen from the combo box in ascending order only.

Person Information System

Person Information

First Name:   
Last Name:   
Title:   
Age:   
Phone:   
Email:

Add Update Delete Reset

Sort Last Name

Filter

Import JSON  Browse

First Name	Last Name	Title	Age	Phone	Email
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khaled	Miss	10	21512951	jana@ha.com
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...

orange™



## TC#8:

**Title:** Filter based on any field.

**Description:** the user can filter the table based on a value he picks based on any field.

**Precondition:** the user should have imported the JSON file already.

### Test Steps:

1. Enter the value to be filtered on in the text box beside the 'Filter' button.
2. Click on 'Filter' button.

The screenshot shows the 'Person Information System' window. On the left, there is a form for adding or updating a person's information with fields for First Name, Last Name, Title (a dropdown menu currently showing 'Mr'), Age, Phone, and Email. Below the form are buttons for 'Add', 'Update', 'Delete', 'Reset', 'Sort', and 'Filter'. The 'Filter' button is highlighted, and the text 'ahmed' is entered in the adjacent text box. On the right, a table displays the current data. The table has columns: First Name, Last Name, Title, Age, Phone, and Email. The data rows are: Mohamed Desoky (Mr, 28, 214125125, Mohamed@l...), jana khaled (Miss, 10, 21512951, jana@ha.com), Ahmed Mohamed (Mr, 22, 0101255343, ahmed@g.c...), New Person (Mr, 25, 010212121, new@new.c...), Update Person (Mr, 30, 219512571, Update@u.c...), and Add Test (Ms, 25, 01222511654, addtest@t.c...).

First Name	Last Name	Title	Age	Phone	Email
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khaled	Miss	10	21512951	jana@ha.com
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	addtest@t.c...

**Expected Result:** The table should show the rows that only contains the value that he have enter based on this case the only record should be shown.

The screenshot shows the same 'Person Information System' window after the filter operation. The table now only contains one record: Ahmed Mohamed (Mr, 22, 0101255343, ahmed@g.c...). The 'Filter' button remains highlighted, and the text 'ahmed' is still in the filter box. At the bottom of the window, there is a new section with an 'Import JSON' button, a text box containing the file path 'C:/Users/badom/workspace/Person Information/myJSON.json', and a 'Browse' button.

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...

## TC#9:

**Title:** Adding an existing row.

**Description:** the user can't enter any duplicated rows.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** there is already a row with the same values that the user wants to add.

### Test Steps:

1. Type in the data in 'First Name', 'Last Name', 'Age', 'Phone', 'Email' and choose the 'Title' from the combo box.
2. Click the 'Add' Button.

The screenshot shows the 'Person Information System' window. On the left, there is a form titled 'Person Information' with fields for First Name (Ahmed), Last Name (Mohamed), Title (Mr), Age (22), Phone (0101255343), and Email (ahmed@g.com). Below the form are buttons for Add, Update, Delete, and Reset. On the right, there is a table with columns: First Name, Last Name, Title, Age, Phone, and Email. The table contains the following data:

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@l...
jana	khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...

An orange logo is visible in the top right corner of the window.

**Expected Result:** A message box appears saying that this row already exists! And doesn't add the row in the table.

The screenshot shows the 'Person Information System' window with the same form and table as the previous image. A message box titled 'Person Information System' is displayed in the center, with a red 'X' icon and the text 'This row already exists!'. Below the message box, there is an 'OK' button. The form fields and table data remain the same as in the previous screenshot.

## TC#10:

**Title:** Update conflict with an existing row.

**Description:** the user can't update the information of specific person into an already existing row.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** there is already a row with the same values that the user wants to update and the user has already selected the row he wants to update.

### Test Steps:

1. Type in the data in 'First Name', 'Last Name', 'Age', 'Phone', 'Email' and choose the 'Title' from the combo box.
2. Click the 'Update' Button.

The screenshot shows the 'Person Information System' application. On the left, there is a form titled 'Person Information' with fields for First Name (Ahmed), Last Name (Mohamed), Title (Mr), Age (22), Phone (0101255343), and Email (ahmed@g.com). Below the form are buttons for Add, Update, Delete, and Reset. On the right, there is a table with columns: First Name, Last Name, Title, Age, Phone, and Email. The table contains the following data:

First Name	Last Name	Title	Age	Phone	Email
Ahmed	Mohamed	Mr	22	0101255343	ahmed@g.c...
Mohamed	Desoky	Mr	28	214125125	Mohamed@i...
Jana	Khaled	Miss	10	21512951	jana@ha.com
New	Person	Mr	25	010212121	new@new.c...
Update	Person	Mr	30	219512571	Update@u.c...
Add	Test	Ms	25	01222511654	add.test@t.c...

An orange logo is visible in the top right corner of the application window.

**Expected Result:** A message box appears saying that this row already exists! And doesn't add the row in the table.

The screenshot shows the 'Person Information System' application with an error message dialog box displayed. The dialog box has a red 'X' icon and the text 'This row already exists!' with an 'OK' button. The background application window shows the same form and table as the previous screenshot, but the 'Update' button is highlighted. Below the table, there are additional controls: a 'Sort' button, a 'Filter' button, and an 'Import JSON' button with a file path 'C:\Users\badorom\workspace\Person Information\myJSON.json' and a 'Browse' button. An orange logo is also visible in the top right corner.

**TC#11:**

**Title:** Entering invalid data type.

**Description:** the user can't enter any numeric values in the 'First Name' or 'Last Name' or any alphabetical values in the 'Age' or 'Phone'.

**Precondition:** the user should have imported the JSON file already.

**Assumption:** the JSON file should be stored as in key and value pairs in the right format or the values won't be able to be shown in table.

**Test Steps:**

1. Enter a numeric value in the 'First Name' or 'Last Name' or any alphabetical values in the 'Age' or 'Phone'.

**Expected Result:** There will be no data enter if user tries to enter invalid data types in the text fields.

**TC#12:**

**Title:** Entering invalid path to a JSON file.

**Description:** the application handles any invalid path to a JSON file and throws an exception based on the case it gets.

**Assumption:** Entered an invalid path or irrelevant data in the 'Import JSON' text field.

**Expected Result:** Exception handled with a message in the console what kind or exception was that and the application continues without crashing.

**TC#13:**

**Title:** Resetting the application to first screen.

**Description:** If the user has entered a lot of inputs and wishes to clear them all there is a button for it.

**Assumption:** Entered data all of the fields.

**Expected Result:** Cleared view of the application as it was in the first place.

**TC#14:**

**Title:** Adding data to the table.

**Description:** If user tried to edit any data in the table it will not be possible

**Assumption:** JSON file has already been loaded.

**Expected Result:** A non-editable version of table loads up with importing the JSON file.

**TC#15:**

**Title:** All the transactions apply in the JSON file.

**Description:** If the user has made as many add, update or delete in the table it will be applied to the JSON file and can be accessible through any editing tool.

**Expected Result:** Changes has been made in the JSON file.