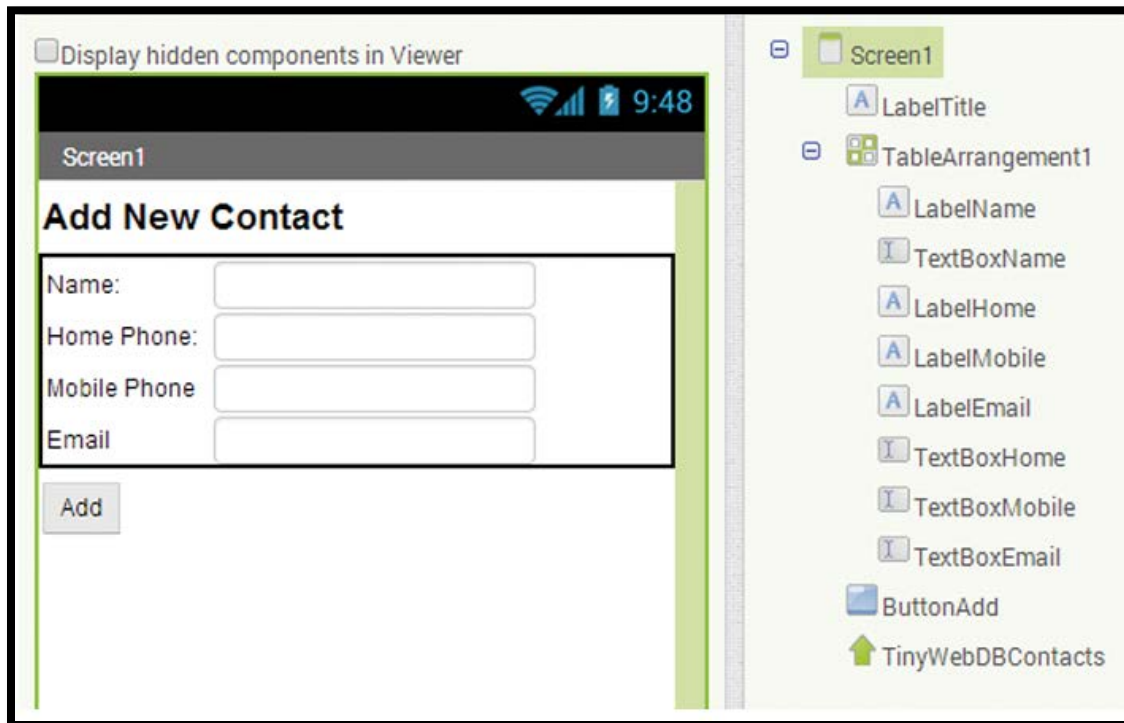


## Storing a List as a Value in a Tag-Value Pair. Part 1

In this App, you will create another Contact application, but this time you will gather the contact's name, home phone, mobile phone, and email. You will make a list of the home phone, mobile phone, and email, and store it as the value for the name, which we will use as the tag.

For this application, it will be important to create each contact's list in a consistent manner. For example, the first element in each list should always be the home phone, the second should be the mobile phone, and the third should always be the email address. This is important so that when we retrieve the values, we can be sure that data is in the correct place.



Step 1: Add a Label and change the Text property to Add New Contact. Change the font size to 20 and mark it as bold.

Step 2: Add a TableArrangement from the Layout Palette. Set the Columns property to 2 and the Rows property to 4 to give the arrangement two columns and four rows.

Step 3: Add four TextBoxes.

- Rename the first TextBoxName and change the Hint property to Enter Name.
- Rename the second TextBoxHomePhone and change the Hint property to Enter Home Phone.
- Rename the third TextBoxMobilePhone and change the Hint property to Enter Mobile Phone.
- Rename the fourth TextBoxEmail and change the Hint property to Enter Email.

Step 4: Add a Button, rename it ButtonAdd, and change the Text property to Add Contact.

Step 5: Add the TinyDB to the project and rename it TinyDBContacts.

Step 6: Program the blocks to add a contact to your TinyDB.

- Open the Blocks Editor, create a variable, ContactInfo, and plug in the create empty list block found in the Lists drawer.
- This will let App Inventor know that ContactInfo will store a list.

Step 7: Now you will program the when ButtonAdd.Click do event handler.

- When the user clicks the Add Button, we want to store the values they typed in for the home phone, mobile phone, and email into a list and store it to the ContactInfo variable.
- Then, we will save the name as the tag and ContactInfo as the value to your TinyDB.
- Go to ButtonAdd and select the when ButtonAdd.Click do block.

Step 8: Hover over the variable name ContactInfo in the initialize global ContactInfo block and select a set global ContactInfo to block.

- Go to the List drawer and select the make a list block, plug that into the ContactInfo block
- Plug the ContactInfo block into the when ButtonAdd.Click do block.

Step 9: Add the TextBoxHomePhone.Text, TextBoxMobile.Text, and the TextBoxEmail.Text blocks into the make a list block.

Step 10: Now store the information to the TinyDB by going to TinyDBContacts and selecting the TinyDBContacts.StoreValue block.

- Plug in the TextBoxName.Text block as the tag value and global ContactInfo block as the valueToStore value.
- You can find the global ContactInfo block by hovering over the variable name ContactInfo in the initialize global ContactInfo block.

Step 11: You can run and test your application now, but you'll notice that you need to clear out the TextBoxes after you save a contact to make your application more user-friendly.

- Set each TextBox's Text property to an empty text block.
- Plug this set of blocks into the when ButtonAdd.Click do event handler at the bottom under the TinyDB.StoreValue block.

