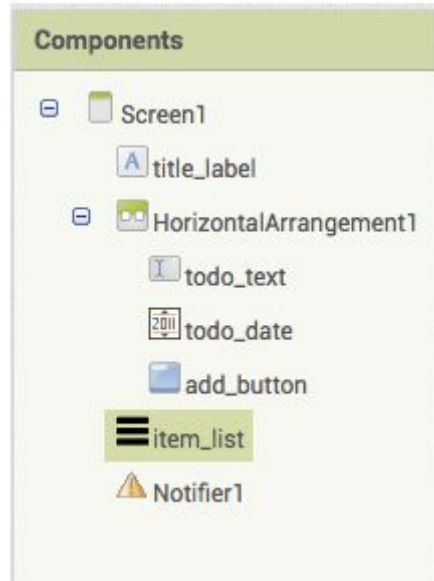


Todo List

Interface Design - Components



Properties

NOTE: All components font should be set to the following: FontTypeface: Monospace

HorizontalArrangement1: Width: Fill Parent, Align Horizontal and Vertical: Center

Todo_date (DatePicker component): Height and Width: 40 pixels, Image: calendaricon.png (upload from folder)

Add_button: height and width: 40 pixels, Image: addicon.png (upload from folder)

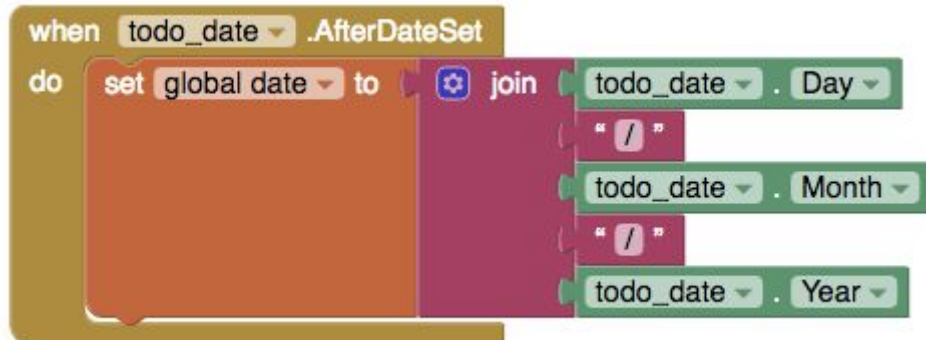
Item_list: Background Color: None, SelectionColor: Light Gray, TextColor: Dary Gray

Blocks

The following blocks of codes setup variables that will store all the data.

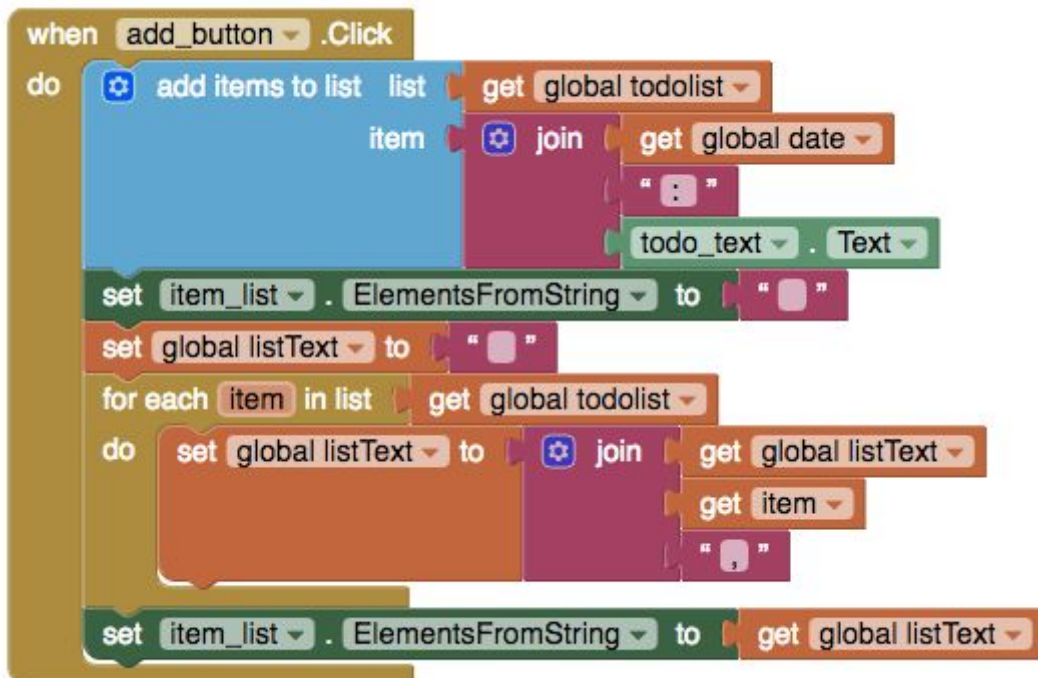


When the user selects the date that work/task must be completed, we must store that date in an appropriate format. We use the 'AfterDateSet' block to capture the date user has selected.



```
when todo_date .AfterDateSet
do
  set global date to
  join
    todo_date . Day
    "/"
    todo_date . Month
    "/"
    todo_date . Year
```

The code for the add button looks complex but once broken down is simple to understand. The purpose of the code is to combine the date and the text entered and add this to the todo list. First we must add the entry to a list then iterate through every item in the list and reload the listview component. This will be explained by your teacher in more detail.

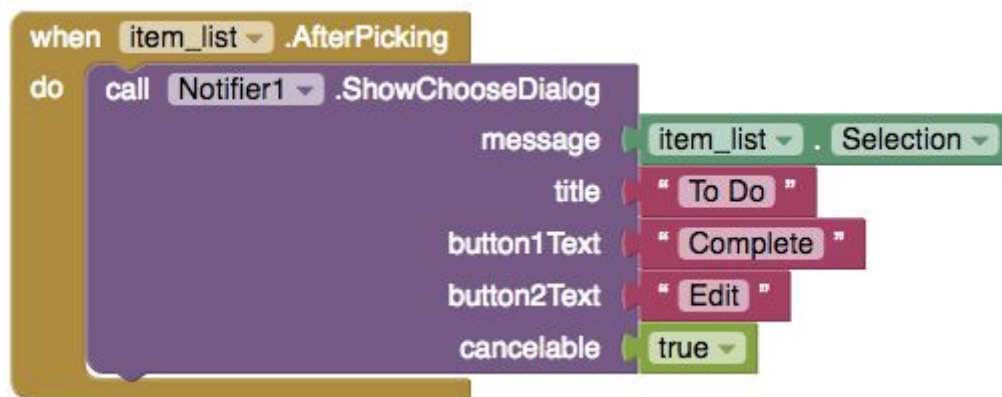


```
when add_button .Click
do
  add items to list list
    item
      get global todolist
      join
        get global date
        ":"
        todo_text . Text
  set item_list . ElementsFromString to ""
  set global listText to ""
  for each item in list
    get global todolist
    do
      set global listText to
        join
          get global listText
          get item
          ","
  set item_list . ElementsFromString to
    get global listText
```

When the user selects an item from the list a popup will appear, providing the user with the following options:

- Complete
- Edit
- Cancel

This is the purpose of the notifier component.



Challenges

- Implement the 'Complete' and 'Edit' buttons (HINT: you will need to use the 'AfterChoosing' block). When the user selects the 'Complete' button it should be removed from the list. This code will be similar to the code from the add button code.



- Add a new feature that will enhance the app.