A of list items

For this first project, we're going to create a webpart that displays a simple responsive table of receipts. Receipts will be stored in a basic SharePoint list, and each row or our webpart will display information specific to a receipt.

I'm going to assume you've already procured and prepared your toolset for creating webparts. If that isn't the case, please refer to Appendix A for a checklist necessary steps and tools you need to have completed and procured before proceeding with this first project. The appendix includes links to various resources and online tutorials.

Scaffolding the project

We begin by create a new webpart project using the usual method:

```
    mkdir receipt-table-webpart
    cd receipt-table-webpart
    yo @microsoft/sharepoint
```

Project options

Yeoman will display a list of questions related to the project. In this case, We'll keep the default name (y), and choose "SharePoint Online only" as a platform. As well, we'll "Use the current folder," and we won't want to allow the tenant admin to deploy the solution immediately (N).

Finally, the components in our solution will not require unique API permissions, and we'll be creating a "WebPart" component, which we will name 'receipt-table-dev'. As description, we'll enter that "it displays a table of receipt." Our solution will not require any JavaScript framework.'

Yeoman will then proceed and scaffold our solution.

Setting up the project's

You should also have git installed.

```
    mkdir receipt-table-webpart
    cd receipt-table-webpart
    yo @microsoft/sharepoint
    git init
```

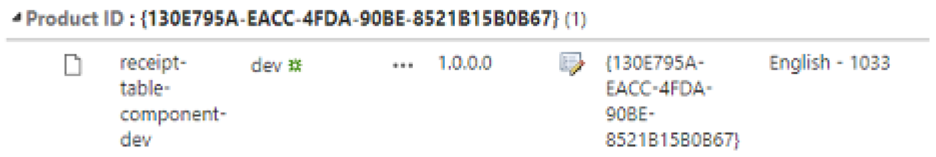
Deploy the web part

Prepare your webpart for deployement by running:

```
    gulp bundle --ship
    gulp package-solution --ship
```

we are now ready to upload our new webpart scaffold to our tenant. If you haven't set up an app library for your tenant yet, you'll need to do so before this next step. I've left detailed instructions in Appendix.

Once your webpart uploaded to the app library, you should see a new item similar to the one below in your app library:



Product ID : {130E795A-EACC-4FDA-90BE-8521B15B0B67} (1)

receipt-table-component-dev	dev	...	1.0.0.0	{130E795A-EACC-4FDA-90BE-8521B15B0B67}	English - 1033

Create a list to store the receipts

Now that our webpart is scaffolded, and uploaded to our tenant, we need to create a new site on your tenant, or, alternatively, we can use an exisiting site. Our site will contain only 3 objects for now:

- Our webpart
- A page that displays our webpart
- A list that contains receipts.

Adding the webpart to the site

Adding adding a new page to the site

Adding the webpart to the page

Creating the list of receipts

For now, our receipt list will contain only one custom column called amount, of type ”currency”

Adding the amount column

Setting up the permissions for the list

We're also going to set up our list so that site members can only view and edit the receipts they've created themselves.

Create a new site: Create a new library:

3

foo: (x,y) =¿ xy ; foo('da','ma'); ¿dama

func2: (x,y) =¿ xxx func2(5) ¿555

fun3: (x,y) =¿ func2(x,x)foo(x,y)

fun3(3,da)=¿3333da

Yet the notion of true and false as atomic variable is a ...it hides what these really are behind...

In programming as in life, one of the often missed is that we only ever the value, true, and false, to decide between one of two paths. That is their only purpose, ever. Booleans, are only useful as arguments to if statements, (if this otherwose that) or as decisions to continue or halt a loop.

So in our toy programming language, I decided to change that. Instead of

ow, functions ca

Where programming meets logic

Phyosovpy, relies on the tenant of predicate logic, typicall expressed as:

The link between logic

For example, consider a function that takes a number as its only argument, and returns the square of that number.

s(4) = 16 ....

Creating dev, testing, and production versions of your webpart

As your webpart evolves, you'll have a version running and being used on the tenant,

Creating a dev version