

## Layout components in GatsbyJS

### How to use Plugins:

If you don't already have a gatsby site you have to create one

- <https://www.gatsbyjs.org/tutorial/part-zero/#install-gatsby-cli>

To use a Plugin for gatsby there are two main steps called installing and configuring:

In this example we will use the typography plugin.

At first you have to install the package.

```
C:\Users\David Andraschko\Desktop\ITP-Projekt\layoutcomponents\layoutComponents.UsePlugin>npm install --save gatsby-plugin-typography react-typography typography
npm WARN eslint-plugin-graphql@2.1.1 requires a peer of graphql@^0.12.0 || ^0.13.0 but none is installed. You must install peer dependencies yourself.
npm WARN express-graphql@0.6.12 requires a peer of graphql@^0.10.0 || ^0.11.0 || ^0.12.0 || ^0.13.0 but none is installed. You must install peer dependencies yourself.
npm WARN graphql-tools@3.1.1 requires a peer of graphql@^0.13.0 but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
+ typography@0.16.18
+ gatsby-plugin-typography@2.2.7
+ react-typography@0.16.18
added 11 packages from 6 contributors and audited 18009 packages in 11.995s
found 0 vulnerabilities
```

After installing the package you have to create a new file in the root of your project.  
This file should be called *gatsby-config.js*

Now you can open this file and add following code:

```
module.exports = {
  plugins: [`gatsby-plugin-typography`],
}
```

This code will select the typography plugin for your own site.

Now start your development server with *gatsby develop*

To test if the installing and configuring was successful you can go to your browser and open your site with localhost:8000

In the developer mode of fire you can see following:

```
<!DOCTYPE html>
<html> <event>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="x-ua-compatible" content="ie=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <script src="/socket.io/socket.io.js"></script>
    <style id="typography.js">
      html{font-family:sans-serif;-ms-text-size-adjust:100%;-webkit-text-size-adjust:100%;}body{margin:0;}article,aside,details,figcaption,figure,footer,header,main,m
      block,audio:not([controls]){display:none;height:0;}progress{vertical-align:baseline}[h1
      skip:objects]:a,a:active,a:hover{outline-width:0;}abbr[title]{border-bottom:none;text-deco
      weight:bolder}dfn{font-style:italic}h1{font-size:2em;margin:.67em 0;}mark{background-co
      align:baseline}sub{bottom:-.25em}sup{top:-.5em}img{border-style:none}svg:not(:root){ov
      40px}hr{box-sizing:content-box;height:0;overflow:visible}button,input,optgroup,select,
      transform:none}tfoot>submit1.button.html tfoe=button1{-webkit-appearance
```

Now you can make some content and style changes.  
You just have to add some code to your index.js file

```
import React from "react"

export default () => (
  <div style={{ margin: `3rem auto`, maxWidth: 600 }}>
<h1>I just installed the Typography plugin</h1>
    <p>
      I can install and configure plugins in gatsby. Now we will
      learn how to create layout component.
    </p>
  </div>
)
```

## How to create layout components:

Add a page to project (page.js)

Now create a new folder at src/components and in this folder a file named layout.js

In you layout.js file you can now define how your page should look:  
Example:

```
import React from "react"

export default ({ children }) => (
  <div style={{ margin: `3rem auto`, maxWidth: 800, padding: `0 1rem` }}>
    {children}
  </div>
)
```

After this you can in import this layout in the sites which should have this layout with:

```
import Layout from "../components/layout"
```

Now you have to remove the <div> </div> tags from your sites except the layout site.

Also you have to add a <Layout> tag after export and a </Layout> tag at the end of your file but before the round bracket which closes the export.

Now you can add a site title and different links in your layout file.