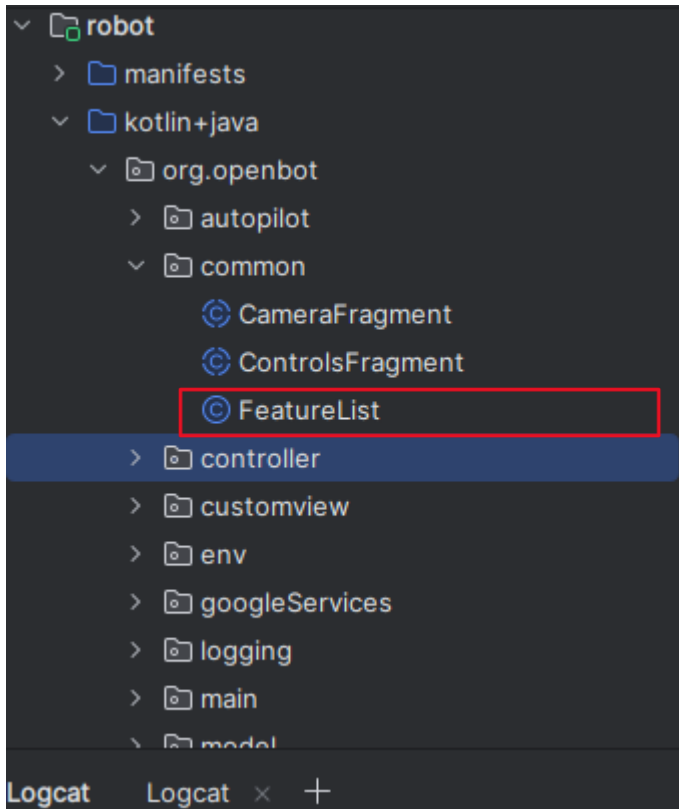
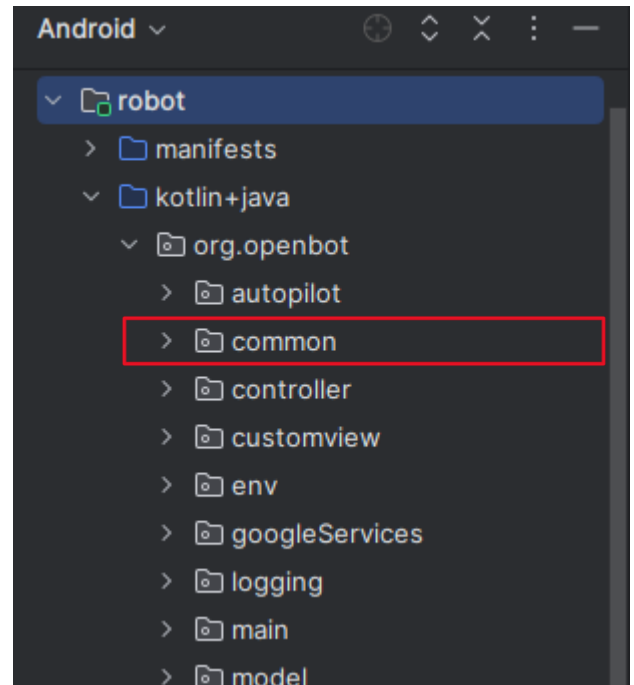
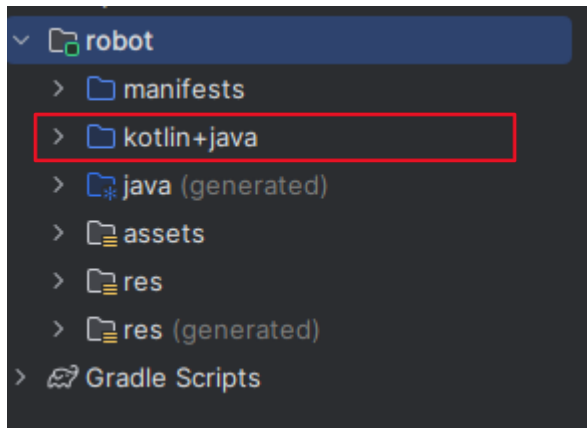


# Documentación.

## Cambiar o agregar imagen en actividad de proyecto en Android Studio



En la raíz del proyecto, nos dirigimos hacia el paquete **kotlin+java**. Posteriormente nos desplazamos a la carpeta **common** por último a la clase llamada **FeatureList**

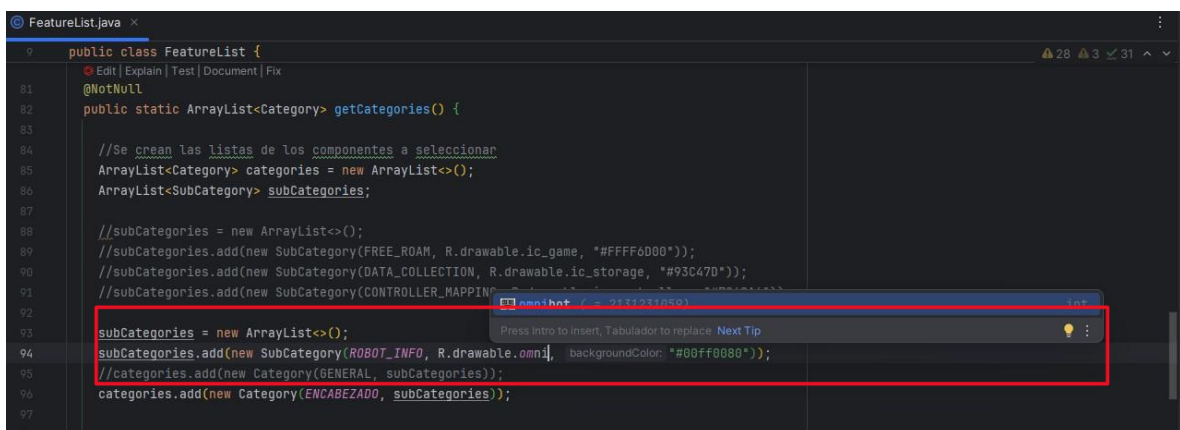


Ahora dentro del código que pertenece a la clase, encontraremos código acerca del contenido de la primera actividad. Lo que nos importa es ir a la línea 94 donde ahí encontraremos la leyenda de la categoría como la imagen con la cual queremos que cargue.



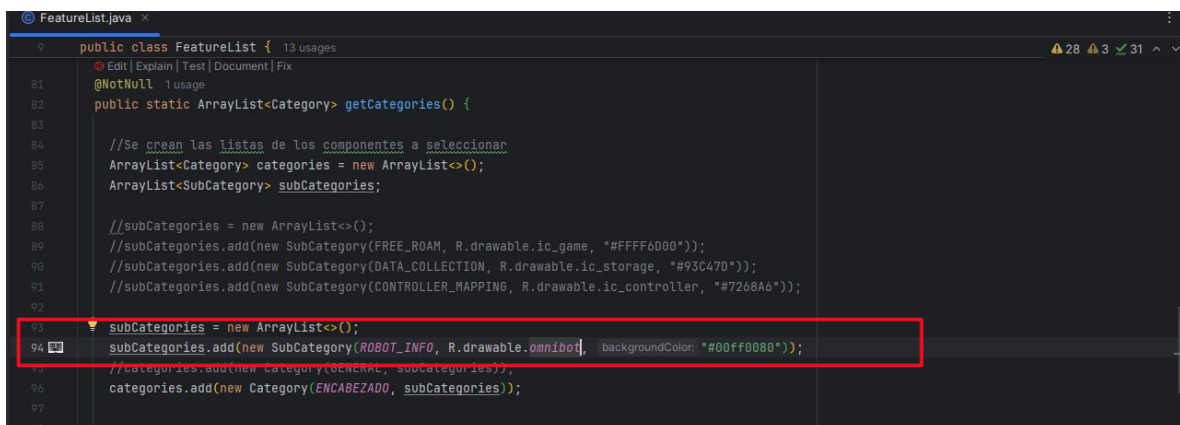
```
9 public class FeatureList { 13 usages
81 @NotNull 1 usage
82 public static ArrayList<Category> getCategories() {
83
84     //Se crean las listas de los componentes a seleccionar
85     ArrayList<Category> categories = new ArrayList<>();
86     ArrayList<SubCategory> subCategories;
87
88     //subCategories = new ArrayList<>();
89     //subCategories.add(new SubCategory(FREE_ROAM, R.drawable.ic_game, "#FFFF6D00"));
90     //subCategories.add(new SubCategory(DATA_COLLECTION, R.drawable.ic_storage, "#93C47D"));
91     //subCategories.add(new SubCategory(CONTROLLER_MAPPING, R.drawable.ic_controller, "#7268A6"));
92
93     subCategories = new ArrayList<>();
94     subCategories.add(new SubCategory(ROBOT_INFO, R.drawable. , backgroundColor: "#00ff0080"));
95     //categories.add(new Category(GENERAL, subCategories));
96     categories.add(new Category(ENCABEZADO, subCategories));
97 }
```

Es sencillo, nos posicionamos justo después de R.drawable y después del punto escribiremos el nombre de nuestra imagen.



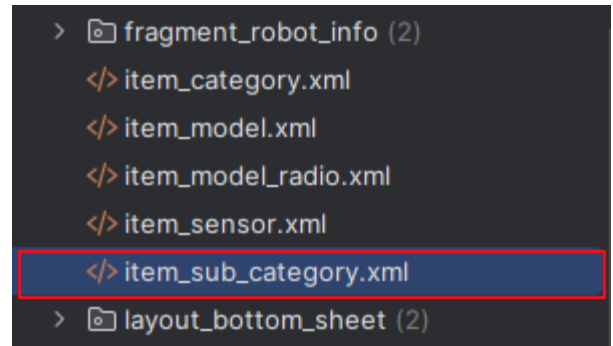
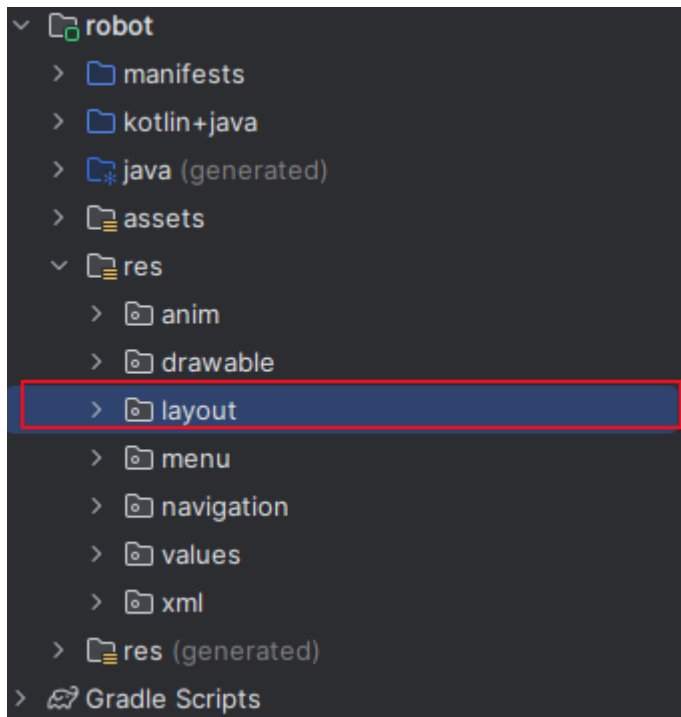
```
9 public class FeatureList {
81 @NotNull
82 public static ArrayList<Category> getCategories() {
83
84     //Se crean las listas de los componentes a seleccionar
85     ArrayList<Category> categories = new ArrayList<>();
86     ArrayList<SubCategory> subCategories;
87
88     //subCategories = new ArrayList<>();
89     //subCategories.add(new SubCategory(FREE_ROAM, R.drawable.ic_game, "#FFFF6D00"));
90     //subCategories.add(new SubCategory(DATA_COLLECTION, R.drawable.ic_storage, "#93C47D"));
91     //subCategories.add(new SubCategory(CONTROLLER_MAPPING, R.drawable.ic_controller, "#7268A6"));
92
93     subCategories = new ArrayList<>();
94     subCategories.add(new SubCategory(ROBOT_INFO, R.drawable.omni, backgroundColor: "#00ff0080"));
95     //categories.add(new Category(GENERAL, subCategories));
96     categories.add(new Category(ENCABEZADO, subCategories));
97 }
```

Recordemos que la imagen debe ya encontrarse en nuestro proyecto, así nos aparecerá de manera rápida y se autocompletará, de otra forma, no estará y tendremos que agregarla.

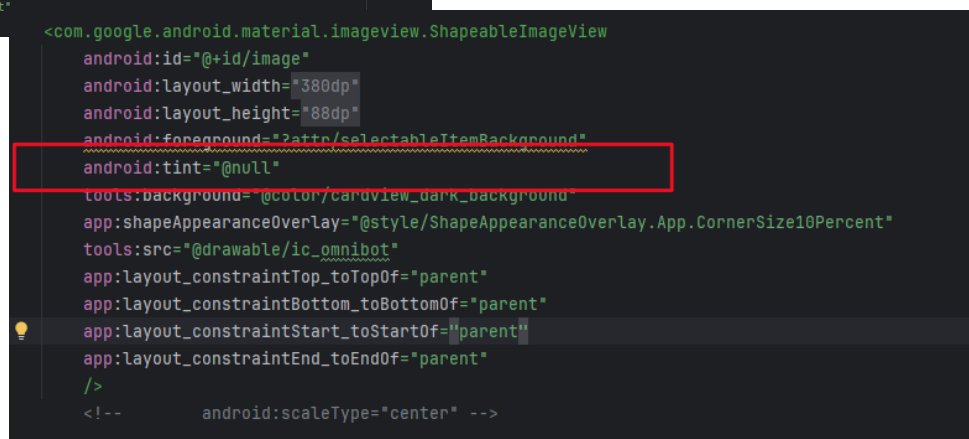
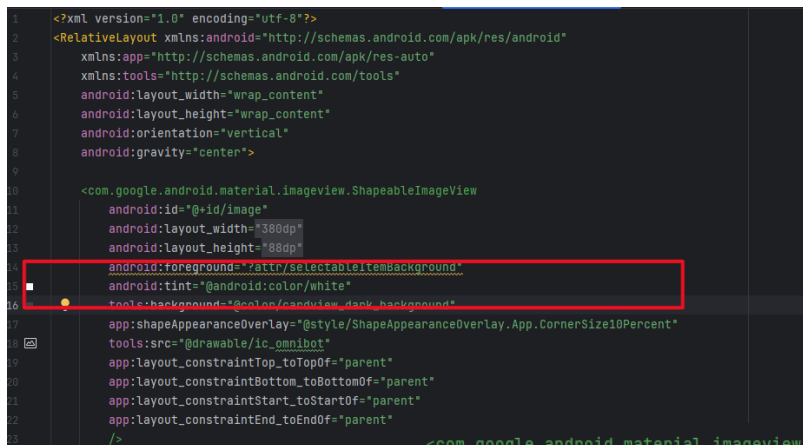


```
9 public class FeatureList { 13 usages
81 @NotNull 1 usage
82 public static ArrayList<Category> getCategories() {
83
84     //Se crean las listas de los componentes a seleccionar
85     ArrayList<Category> categories = new ArrayList<>();
86     ArrayList<SubCategory> subCategories;
87
88     //subCategories = new ArrayList<>();
89     //subCategories.add(new SubCategory(FREE_ROAM, R.drawable.ic_game, "#FFFF6D00"));
90     //subCategories.add(new SubCategory(DATA_COLLECTION, R.drawable.ic_storage, "#93C47D"));
91     //subCategories.add(new SubCategory(CONTROLLER_MAPPING, R.drawable.ic_controller, "#7268A6"));
92
93     subCategories = new ArrayList<>();
94     subCategories.add(new SubCategory(ROBOT_INFO, R.drawable.omnibot, backgroundColor: "#00ff0080"));
95     //categories.add(new Category(GENERAL, subCategories));
96     categories.add(new Category(ENCABEZADO, subCategories));
97 }
```

Ahora podemos que se cargo correctamente y ahora solo queda volver a lanzar la aplicación. En caso de que no se logre ver la imagen o quieras cambiar el tamaño de las dimensiones tenemos que ir al **layout**, para después ir al XML llamado **item\_sub\_category.xml**.



Sustituir la línea 15 por esto: `android:tint="@null"`



Una vez esto, puede cambiar la imagen en cualquier momento.

