

本次作业运行代码只需在终端运行 `roslaunch learning_urdf display_mbot_base.launch` 实现从零创建一个 urdf 模型，步骤如下：

1、首先在工作空间的 `src` 文件夹下创建功能包

终端输入 `catkin_create_pkg learning_urdf urdf xacro`

2、在 `learning_urdf` 功能包下手动创建 `urdf`（存放 urdf 或者 xacro）、`meshes`（存放 solidworks 文件）、`launch`（launch 启动文件）和 `config`（rviz 配置文件）四个文件夹

3、在 `urdf` 文件夹中，touch `mbot_base.urdf`

4、在 `mbot_base.urdf` 中，写入如下 xml 格式的代码

```
<?xml version="1.0" ?>
```

```
<robot name="mbot">
```

```
  <link name="base_link">
```

```
    <visual>
```

```
      <origin xyz="0 0 0" rpy="0 0 0" />
```

```
      <geometry>
```

```
        <box size="0.3 0.3 0.02"/>
```

```
      </geometry>
```

```
      <material name="yellow">
```

```
        <color rgba="1 0.4 0 1"/>
```

```
      </material>
```

```
    </visual>
```

```
  </link>
```

```
  <joint name="left_wheel_joint" type="continuous">
```

```
    <origin xyz="-0.11 0.15 0" rpy="0 0 0"/>
```

```
    <parent link="base_link"/>
```

```
    <child link="left_wheel_link"/>
```

```
    <axis xyz="0 1 0"/>
```

```
  </joint>
```

```
  <link name="left_wheel_link">
```

```
    <visual>
```

```
      <origin xyz="0 0 0" rpy="1.5707 0 0" />
```

```
      <geometry>
```

```
        <cylinder radius="0.045" length="0.02"/>
```

```
      </geometry>
```

```
      <material name="white">
```

```
        <color rgba="1 1 1 0.9"/>
```

```
      </material>
```

```
    </visual>
```

```
  </link>
```

```
  <joint name="right_wheel_joint" type="continuous">
```

```
    <origin xyz="-0.11 -0.15 0" rpy="0 0 0"/>
```

```
    <parent link="base_link"/>
```

```
    <child link="right_wheel_link"/>
```

```
    <axis xyz="0 1 0"/>
```

```
  </joint>
```

```
<link name="right_wheel_link">
  <visual>
    <origin xyz="0 0 0" rpy="1.5707 0 0" />
    <geometry>
      <cylinder radius="0.045" length = "0.02"/>
    </geometry>
    <material name="white">
      <color rgba="1 1 1 0.9"/>
    </material>
  </visual>
</link>
```

```
<joint name="front_caster_joint" type="continuous">
  <origin xyz="0.12 0 -0.025" rpy="0 0 0"/>
  <parent link="base_link"/>
  <child link="front_caster_link"/>
  <axis xyz="0 1 0"/>
</joint>
```

```
<link name="front_caster_link">
  <visual>
    <origin xyz="0 0 0" rpy="0 0 0"/>
    <geometry>
      <sphere radius="0.02" />
    </geometry>
    <material name="black">
      <color rgba="0 0 0 1"/>
    </material>
  </visual>
</link>
```

```
<link name="kinect_link">
  <visual>
    <origin xyz="0 0 0" rpy="0 0 1.5708"/>
    <geometry>
      <mesh filename="package://learning_urdf/meshes/kinect.dae" />
    </geometry>
  </visual>
</link>
```

```
<joint name="kinect_joint" type="fixed">
  <origin xyz="0.12 0 0.045" rpy="0 0 0"/>
  <parent link="base_link"/>
  <child link="kinect_link"/>
</joint>
```

</robot>

5、在 launch 文件夹中，touch display\_mbot\_base.launch，写入以下代码：

<launch>

```
<param name="robot_description" textfile="$(find learning_urdf)/urdf/mbot_base.urdf"/>
```

```
<!-- 运行 joint_state_publisher 节点，发布机器人的关节状态 -->
```

```
<node name="joint_state_publisher_gui" pkg="joint_state_publisher_gui"
type="joint_state_publisher_gui"/>
```

```
<!-- 运行 robot_state_publisher 节点，发布 tf -->
```

```
<node name="robot_state_publisher" pkg="robot_state_publisher" type="robot_state_publisher"/>
```

```
>
```

```
<!-- 运行 rviz 可视化界面 -->
```

```
<node name="rviz" pkg="rviz" type="rviz" args="-d $(find
learning_urdf)/config/mbot_urdf.rviz" required="true"/>
```

</launch>

6、在工作空间下的终端中运行 catkin\_make 进行编译

7、终端下运行 roslaunch learning\_urdf display\_mbot\_base.launch

8、在打开的 rviz 中，添加 robotmodel，并将 fixed frame 修改为 base\_link，就可以显示 urdf 可视化结果，在 rviz 中保存 mbot\_urdf.rviz 文件到 config 中。下次打开直接就是 urdf 模型

