Build-in to Universal Render Pipeline Conversion

Cheat Sheet

	Builtin	URP
Shader code section	CGPROGRAM CGINCLUDE ENDCG	HLSLPROGRAM HLSLINCLUDE ENDHLSL
Pass Tags	"LightMode" = "ForwardBase"	"LightMode" = "UniversalForward"
Multi compile	#pragma multi_compile_fwdbase	<Not required>

Includes	#include "UnityCG.cginc"
	#include "UnityLightingCommon.cginc"
		#include "Packages/com.unity.render-pipelines.universal/ShaderLibrary/Core.hlsl"
		#include "Packages/com.unity.render-pipelines.universal/ShaderLibrary/Lighting.hlsl"

Transform space conversion	UnityObjectToClipPos UnityWorldToClipPos UnityWorldSpaceViewDir UnityObjectToWorldNormal	TransformWorldToHClip TransformWorldToHClip TransformWorldToViewDir TransformObjectToWorldNormal
Parameter container naming	appdata v2f	Attributes Varyings

Get Main Light	half3 lightColor = _LightColor0.rgb;
	half3 lightDirection = _WorldSpaceLightPos0;
	half shadowAttenuation = LIGHT_ATTENUATION(IN);
		Light mainLight = GetMainLight(input.shadowCoords);
		half3 lightColor = mainLight.color;
		half3 lightDirection = mainLight.direction;
		half shadowAttenuation = mainLight.shadowAttenuation;

Shadow Coordinates	TRANSFER_SHADOW(OUT);	output.shadowCoords = TransformWorldToShadowCoord(output.positionWS);

Sample Depth Pass	sampler2D_float DepthPassTexture;
	...
	float4 screenPosition = ComputeScreenPos(positionCS);
	COMPUTE_EYEDEPTH(screenPosition);
	float sceneZ = tex2Dproj(DepthPassTexture, screenPosition);
	float thisZ = screenPosition.z;
		#include "Packages/com.unity.render-pipelines.universal/ShaderLibrary/DeclareDepthTexture.hlsl"
		...
		float2 screenPosition = input.positionCS.xy / _ScaledScreenParams.xy;
		float rawDepth = SampleSceneDepth(params.projectedPosition.xy / params.projectedPosition.w);
		float sceneZ = (unity_OrthoParams.w == 0) ? LinearEyeDepth(rawDepth, _ZBufferParams) : LinearDepthToEyeDepth(rawDepth);
		float thisZ = LinearEyeDepth(params.positionWS.xyz, GetWorldToViewMatrix());

Sample Opaque Pass Texture	sampler2D ColorPassTexture;
	...
	float4 screenPosition = ComputeScreenPos(positionCS);
	half4 color = tex2Dproj(ColorPassTexture, screenPosition);
		#include "Packages/com.unity.render-pipelines.universal/ShaderLibrary/DeclareOpaqueTexture.hlsl"
		...
		float2 screenPosition = input.positionCS.xy / _ScaledScreenParams.xy;
		SampleSceneColor(screenPosition);