Chapter 7: 🛒 `FrameBuffer` [🍇🍅🥚🍗]

1. 🛠 `vkCreateFramebuffer()`

- *https://vkdoc.net/man/vkCreateFramebuffer*
- 📜 *REY_DOCs*
  - *Copy Paste* `amVK_RenderPass.hh` *Current Implementation & Change it as needed*
    - *Trust me, this is the most fun way of doing this, xP*

2. 🎁 `VkFramebufferCreateInfo()`

- *https://vkdoc.net/man/VkFramebufferCreateInfo*
  - `.flags` ☐ `0`
    - *https://vkdoc.net/man/VkFramebufferCreateFlagBits* | *ivirtex-github*
    - 🚩 `VK_FRAMEBUFFER_CREATE_IMAGELESS_BIT` *[ImageLess FrameBuffer]*
  - `.renderPass` ▨ 👩‍🏫
  - `.pAttachments` 🔗 *SubChapter 3*
  - `.width`
  - `.height`
  - `.layers`
- 📜 *REY_DOCs*
  - *Start With basic copy paste of* `amVK_RenderPass.hh` *:-*
- 📽 `So far, The result` *amVK_RenderPassFBs.hh*

3. `VkImageView .pAttachments`

- *https://vkdoc.net/man/VkImageView*
  - *For Now, We are gonna choose 1 VkImageView per FrameBuffer*
- `</> TheCode`

```
void amVK_RenderPassFBs::CreateFrameBuffers(void) {
    if (this->SC_IMGs->called_GetSwapChainImagesKHR == false)   {
        this->SC_IMGs->       GetSwapChainImagesKHR();          }
    if (this->SC_IMGs->called_CreateSwapChainImageViews == false)   {
        this->SC_IMGs->       CreateSwapChainImageViews();          }

    VkExtent2D imageExtent = this->SC_IMGs->active_ImageExtent();
        this->CI.width  = imageExtent.width;
        this->CI.height = imageExtent.height;

    this->amVK_1D_RP_FBs.reserve(this->SC_IMGs->amVK_1D_SC_IMGs.n);

    REY_Array_LOOP(this->amVK_1D_RP_FBs, k) {
        this->CI.attachmentCount = 1;
        this->CI.pAttachments = &(this->SC_IMGs->amVK_1D_SC_IMGViews[k]);

            #define VK_DEVICE this->RP->D->vk_Device

        VkResult return_code = vkCreateFramebuffer(VK_DEVICE, &CI, nullptr, &this->amVK_1D_RP_FBs[k]);
        amVK_return_code_log( "vkCreateFramebuffer()" );
    }
}
```

- 📽 `So far, The result` *amVK_RenderPass.cpp#L34-L55*