Need to first have the base website for each movie (could be manually copy paste), dowloading 10 images for 10 scenese. Also, best not scrape 4k images cause it takes too long. Even for non4k image, size is (808, 1920, 3), so better resize the images before features extraction

to find the sample image page, go to https://animationscreencaps.com/ (https://animationscreencaps.com/) and pick a movie, click a rondom image and right click and select copy image path. The path should be sth like this, and replace the number before '.jpg' could get different caps in this movie. Typically it is ordered. https://i0.wp.com/caps.pictures/201/4k-spidermaninto/full/4k-spidermaninto-animationscreencaps.com-185.jpg?zoom=1.25&strip=all (https://i0.wp.com/caps.pictures/201/4k-spidermaninto/full/4k-spidermaninto-animationscreencaps.com-185.jpg?zoom=1.25&strip=all)

In [15]:
```
from bs4 import BeautifulSoup
import os
import requests

# saving all data into the scenes folder
base = 'data\\scenes'
```

create a folder named the movie's name

every image's name's format: movie'name (same with the folder's name)-Scene-scene number-cap number.jpg

E.g toy_story-Scene-1-0.jpg

toy_story-Scene-1-1.jpg

toy_story-Scene-1-2.jpg

In [16]:
```
toy_story_4_base = 'https://i1.wp.com/caps.pictures/201/9-toystory4/full/toystory
```

In [17]:
```
scenes = [x*50+3 for x in range(30)]
# *100 represent how many pics between each scene, range(10) 10 represent 30 scen
# +10 represent the begining of your first scene
```

In [18]:
```
def scratch(movie_base, movie_name):
    movie_dir = os.path.join(base, movie_name)
    if os.path.isdir(movie_dir) == False:
        os.mkdir(movie_dir)
    for start_i in range(len(scenes)):
        for i in range(3):
            # 3 represent 3 caps per scene
            img_index = scenes[start_i] + i
            img_path = movie_base.format(num=img_index)
            img_filename = movie_name + '-Scene-' + str(start_i) + '-' + str(i) +
            out_file = os.path.join(movie_dir, img_filename)
            temp = requests.get(img_path, out_file)
            display(out_file)
            if not os.path.exists(out_file):
                with open(out_file, 'wb') as f:
                    f.write(temp.content)
```

In [19]:
```
scratch(toy_story_4_base, 'toy_story_4')
```

```
'data\\scenes\\toy_story_4\\toy_story_4-Scene-0-0.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-0-1.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-0-2.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-1-0.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-1-1.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-1-2.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-2-0.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-2-1.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-2-2.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-3-0.jpg'

'data\\scenes\\toy_story_4\\toy_story_4-Scene-3-1.jpg'
```

In [29]:
```
'data\\scenes\\toy_story_4\\toy_story_4-Scene-25-0.jpg'
```

Out[29]:
```
'25'
```

In [ ]:
```
```