step6.py

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# ========================================
# Find and print the best movie per category
# ========================================

with open('250.imdb', 'r', encoding='utf-8') as f:

    # For each category, I keep the best rating
    # Mapping { key: value } where   key = string
    #                              value = int
    categories = {} # Nothing at the start

    for line in f:

        if line.startswith('#'): # Not interested
            continue

        # Get the fields as a list of strings
        fields = line.split('|')

        # Rename the fields, cuz I prefer, and convert them
        rating = float(fields[1])
        genres = fields[-2].lower().split(',') # List of strings also

        for genre in genres:
            genre = genre[:6] # Cheating
            old_rating = categories.get(genre, 0.0) # No KeyError

            if rating > old_rating: # found a better one
                categories[genre] = rating


    # Print the categories
    for genre,rating in categories.items():
        print("The best movie for",genre,"has rating:",rating)
```

U:--- step6.py All L1 (Python +2 NMN) 19:21 1.08
Use +,-,0 for further adjustment

step7.py

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# ========================================
# Find and print the best movie per category
# ========================================

with open('250.imdb', 'r', encoding='utf-8') as f:

    # For each category, I keep the best rating
    # Mapping { key: value } where   key = string
    #                              value = (int,string)
    categories = {} # Nothing at the start

    for line in f:

        if line.startswith('#'): # Not interested
            continue

        # Get the fields as a list of strings
        fields = line.split('|')

        # Rename the fields, cuz I prefer, and convert them
        rating = float(fields[1])
        title = fields[-1].strip() # Clean the title
        genres = fields[-2].lower().split(',') # List of strings also

        for genre in genres:
            genre = genre[:6]
            old_rating,old_title = categories.get(genre, (0.0,'') ) # No KeyError

            if rating > old_rating: # found a better one
                categories[genre] = (rating, title)


    # Print the categories
    for genre,value in categories.items():
        print("The best movie for",genre,'\n\tis"',value[1],'"\n\tand has rating:',value[0])
```

U:--- step7.py All L1 (Python +2 MMN) 19:20 1.25
Use +,-,0 for further adjustment