*"""  
CP1404/CP5632 Practical  
State names in a dictionary  
File needs reformatting  
"""*# Already was formatted according to PEP 8 convention  
CODE\_TO\_NAME = {"QLD": "Queensland", "NSW": "New South Wales", "NT": "Northern Territory", "WA": "Western Australia",  
 "ACT": "Australian Capital Territory", "VIC": "Victoria", "TAS": "Tasmania"}  
# Old printout  
# print(CODE\_TO\_NAME)  
  
# New printout  
for state\_code, state\_name in CODE\_TO\_NAME.items():  
 print("{:^3} is {}".format(state\_code, state\_name))  
  
state\_code = input("Enter short state: ").upper()  
while state\_code != "":  
 if state\_code in CODE\_TO\_NAME:  
 print(state\_code, "is", CODE\_TO\_NAME[state\_code])  
 else:  
 print("Invalid short state")  
 state\_code = input("Enter short state: ").upper()

COLOUR\_DICT = {"black": "#000000", "blue1": "#0000ff", "blueviolet": "#8a2be2", "chocolate": "#d2691",  
 "coral1": "#ff7256", "cornsilk1": "#fff8dc", "darkgoldenrod1": "#ffb90f", "darkgreen": "#006400",  
 "darkorange": "#ff8c00", "firebrick": "#b22222"}  
  
colour\_selection = input("Please enter a colour name (with no spaces >>> ").lower()  
while colour\_selection != "":  
 try:  
 print(COLOUR\_DICT[colour\_selection])  
 except KeyError:  
 print("Colour selection must appear in the list {}".format(COLOUR\_DICT.keys()))  
 colour\_selection = input("Please enter another colour name >>> ").lower()

placeholder\_input\_text = str(input("Please input a string to count occurrence of each word >>> "))  
# Instantiate new dictionary for words  
word\_dict = {}  
# Splits user string input into list of each word  
words = (placeholder\_input\_text.split())  
# Instantiate new list to store the length of each word  
word\_length = []  
  
# Cycle through each word in the words list.  
for word in words:  
 # First appearance of each word counts places the word in the dictionary and counts it once  
 if word not in word\_dict.keys():  
 word\_dict[word] = 1  
 # Else the counter for the occurrence of the word is incremented  
 else:  
 word\_dict[word] += 1  
 # The length of each word is added to the word\_length list  
 word\_length.append(len(word))  
  
# For each item (tuple) in the sorted dictionary, print the word and its occurrence count  
for item in sorted(word\_dict.items()):  
 print(f"{item[0]:<{max(word\_length) + 1}}{item[1]}")

def main():  
 emails\_dict = {}  
 email\_lengths = []  
 name\_lengths = []  
 new\_email = str(input("Email: "))  
  
 while new\_email != "":  
 name = get\_name\_from\_email(new\_email)  
 emails\_dict[new\_email] = name  
 name\_lengths.append(len(name))  
 email\_lengths.append(len(new\_email))  
 new\_email = str(input("Enter another email: "))  
  
 # Prints sorted email list  
 for item in emails\_dict.items():  
 print(f"{item[1]:<{max(name\_lengths)}} ({item[0]:<{max(email\_lengths)}})")  
  
  
def get\_name\_from\_email(new\_email):  
 email\_split\_by\_dots = new\_email.split(".")  
 email\_further\_split\_by\_ats = email\_split\_by\_dots[1].split("@")  
 name = email\_split\_by\_dots[0].title() + " " + email\_further\_split\_by\_ats[0].title()  
  
 is\_name = str(input("Is your name {}? (y/n): ".format(name)))  
 if is\_name == "n":  
 name = str(input("Name: "))  
 return name  
  
  
main()