

주어진 문자열이 팰린드롬인지 확인하라. 대소문자를 구분하지 않으며, 영문자와 숫자만을 대상으로 한다.

Input = "A man, a plan, a canal: Panama"

Output = true

1.리스트 활용

```
def isPalindrome(s: str) -> bool:
    # 문자열의 필요없는 문자들 삭제
    strs = []
    for char in s:
        if char.isalnum():
            strs.append(char.lower())
    # 팰린드롬 확인
    while len(strs) > 1:
        if strs.pop(0) != strs.pop():
            return False
    return True
```

▸ 기본적인 리스트를 이용한 방법이다.

2.데크 활용

```
def isPalindrome(s: str) -> bool:
    # 데크 활용
    strs: Deque = collections.deque()
    for char in s:
        if char.isalnum():
            strs.append(char.lower())
    while len(strs) > 1:
        if strs.popleft() != strs.pop():
            return False
    return True
```

▸ 데크를 활용해서 pop(0)의 O(n)을 pop left()의 O(1)로 최적화 했다.

3.슬라이싱, 정규식 활용

```
def isPalindrome(s: str) -> bool:
    s = s.lower()
    s = re.sub('[^a-z0-9]', '', s)
    return s == s[::-1]
```

▸ 정규식을 활용해서 필요없는 문자를 걸러내고, 슬라이싱을 통해서 비교한다.

▸ 파이썬다운 풀이다.

▸

