

Power of Two

Given an integer n , return true if it is a power of two. Otherwise, return false.

An integer n is a power of two, if there exists an integer x such that $n == 2^x$.

Example 1:

- **Input:** $n = 1$
- **Output:** true
- **Explanation:** $2^0 = 1$

Example 2:

- **Input:** $n = 16$
- **Output:** true
- **Explanation:** $2^4 = 16$

Example 3:

- **Input:** $n = 3$
- **Output:** false

Constraints:

- $-2^{31} \leq n \leq 2^{31} - 1$

شرح الحل:

لتحديد ما إذا كان العدد n هو قوة للعدد ٢، يمكننا استخدام الخصائص التالية:

- العدد الذي يمثل قوة للعدد ٢ يمتلك تمثيل ثنائي يحتوي على ١ واحد فقط (مثل: ١، ٢، ٤، ٨، ١٦، ... الخ).
- طريقة بسيطة للتحقق هي باستخدام العملية AND بين n و $(n - 1)$. إذا كانت النتيجة تساوي ٠، فهذا يعني أن n هو قوة للعدد ٢.

```
def isPowerOfTwo(n: int) -> bool
```

```
# نتحقق من أن العدد أكبر من ٠ وأن العدد & (العدد - ١) == ٠
```

```
return n > 0 and (n & (n - 1)) == 0
```

شرح الكود:

١. التحقق الأول: نتأكد أن n أكبر من ٠، لأن الأعداد السالبة والأعداد الصفرية ليست قوى للعدد ٢.

٢. التحقق الثاني: نستخدم العملية $(n - 1) \& n$ ، إذا كانت النتيجة ٠، فهذا يعني أن n هو قوة للعدد ٢.

○ على سبيل المثال، إذا كان $n = 16$ ، فالتمثيل الثنائي لـ ١٦ هو 10000، و $(n - 1)$ سيكون

01111. باستخدام AND، ستكون النتيجة 00000، مما يعني أن ١٦ هو قوة للعدد ٢.