ASSIGNMENT-2

Using continue can make loops more **readable and efficient** in certain cases:

1. **Skipping Unwanted Iterations**
   When processing data, you might want to **skip invalid or unnecessary values** without breaking the loop.
   Instead of nesting all valid logic inside an if block, continue lets you **filter out unwanted cases early**, keeping the code clean.

2. **Avoiding Deep Nesting**
   In loops with multiple conditions, using continue can help **reduce indentation** and improve readability.

3. **Skipping Errors in Data Processing**
   When reading files or parsing data, continue can **skip faulty entries** instead of breaking the loop.

When to Avoid Using continue

- If skipping cases frequently, **consider filtering data before looping**.
- If continue makes logic hard to follow, **a different loop structure (like while) might be better**.
- Avoid excessive continue statements, as they may lead to **unintended skips**.