IPO, Flowchart & Code Guidelines

Simplify inputs in diagrams

Keep names short and clear.

Before:

```
Price of product in pesos
```

After:

```
input price_product
```

State processes as the actual formulas

Write the formula exactly as you will use it in code.

Before:

```
Divide price of product in pesos by exchange rate
```

After:

```
dollars = price_product / exchange_rate
```

Keep flowchart names concise

Remove extra words if the meaning is already clear.

Before:

```
Input Partial 1 grade (partial_1)
```

After:

```
input partial_1
```

Apply the same simplification to IPO diagrams

IPO should be as clean and direct as your flowchart. Avoid long sentences or repeated information.

Clarify the origin of variables

Show if values are given by the user or fixed in the program.

User input:

```
input exchange_rate
```

Fixed in code:

```
exchange_rate = 18.81
```

(Place in "Information" section)

Avoid double naming

Keep one name for a variable.

Before:

```
pesos (price)
```

After:

```
input pesos
```

Names should be consistent

If you use a name in Input, use the exact same name in Process.

Input:

```
input pesos
```

Process:

```
usd = pesos / exchange_rate
```

Never use uppercase for variable names (except math symbols)

Use lowercase for variable names. Allowed: F, C in formulas only (if not used for anything else).

Prefer:

```
final_grade
```

Avoid:

```
FinalGrade
```

Use underscores for multiple words

Prefer one word when possible. If you need more than one, connect them with underscores.

Prefer:

```
price_product
final_grade
```

Avoid:

```
price product
priceProduct
```

if someone insists on using camelCase, `priceProduct` is still better than `PriceProduct` because it keeps the first letter lowercase, which is closer to the variable naming convention we're following.

But for our guidelines, we'd still prefer:

```
price_product
```

over camelCase, so it's consistent with Python's typical style and easier to read for beginners.

Use keywords for all inputs and outputs

Inputs:

```
input partial_1
ask partial_1
read partial_1
```

Outputs:

```
print final_grade
show final_grade
display final_grade
```

Indicate when a value is hardcoded

Hardcoded = value written directly into the code, not entered by the user or part of a formula.

Hardcoded data:

```
exchange_rate = 18.81
```

(Place in "Information" section)

Formula constant:

```
F = celsius * (9/5) + 32
```

(OK in Process)