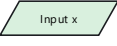
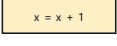
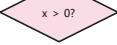


Java Programming Flowchart: Cheat Sheet

Symbols · control structures · Java mapping · common mistakes

SYMBOLS

SHAPE	NAME	JAVA
	Terminal: start/end	main begins/ends
	Input/Output	sc.nextInt(), println
	Process	= ++ += math
	Decision: 2 exits (T/F)	if, loop tests
	On-page connector	joins lines
	Off-page connector	continues on next page
	Predefined process	method call
	Preparation (optional)	for setup

THREE CONTROL STRUCTURES

Sequence: steps in order, top → bottom. No diamonds.

Selection: a diamond splits into True/False paths that **join again**. if, if-else, else if, switch.

Repetition: an arrow goes **back up**. while, for, do-while.

DRAWING RULES

- One Start, at least one End. Every path must reach End.
- Flow is top → bottom, left → right.
- Label both diamond exits: True/False or Yes/No.
- One action per symbol and short text.
- No crossing lines: use connectors.

SELECTION IN JAVA

```
if (n < 0) { n = -n; }           // if: one path
```

```
if (score >= 75) println("Pass"); // if-else:  
else                println("Fail"); // exactly one runs
```

```
if (s >= 90) ...                // else-if ladder:  
else if (s >= 80) ...           // False -> next diamond  
else ...                        // first True wins
```

switch: one diamond per case, default = last False path. No break means it **falls through** to the next case.

LOOPS IN JAVA

```
int i = 1;                // initialize (before loop)  
while (i <= 5) {           // test = diamond  
    println(i);           // body  
    i++;                  // update, then back to test  
}
```

```
for (int i = 1; i <= 5; i++) { ... } // same shape
```

```
do {                      // body FIRST  
    ...  
} while (x < 5);           // test LAST (note the ;)
```

LOOP	TEST	MIN RUNS
while	pre-test (before body)	0
for	pre-test (before body)	0
do-while	post-test (after body)	1

- **Counter-controlled:** known count (for).
- **Sentinel-controlled:** stop on a special value, e.g. -1.
- **break:** arrow straight to the loop exit.
- **continue:** skip to the update/test.

CODE → FLOWCHART IN 6 STEPS

- Start of main → **Start**
- sc.next...() / println → **parallelogram**
- =, ++, math → **rectangle**
- if / loop condition → **diamond**
- Loop → arrow back to the diamond
- End of main → **End**

COMMON MISTAKES

- Using a rectangle for println. It's Output (parallelogram).
- Unlabeled diamond exits.
- sum = 0 inside the loop (it resets every pass).
- No update → infinite loop.
- Loop arrow going back to Start instead of the diamond.