TurboCAD PPM — One-Page Cheat Sheet

Script Skeleton

```
// 1) Units (optional but recommended)
Units(1[in]);

// 2) Declare Parameters
L = Parameter("Length", 5, LINEAR, Interval(0.1, 100));
A = Parameter("Angle", 45, ANGULAR, Set(0, 45, 90));
clr = Parameter("Color", 0xff, COLOR);
P0 = ParameterPoint(0, 0, 0); // user-pickable point

// 3) Compute helpers
x = L/2;

// 4) Build geometry
r = Rectangle(L, L/2);
solid = Thickness(r, 0.75); // extrude board 3/4"
solid = SetProperties(solid, "PenColor" = clr);

// 5) Optional insertion reference
rp = RefPoint(0, 0, 0);

// 6) Output (one or many objects; multiple Output() calls allowed)
Output(solid, rp);
```

Parameters & Validators

- **Types**: `LINEAR`, `ANGULAR`, `COLOR`, `FONT`, `TEXT`, `ParameterPoint(x,y,z)`
- **Validators**: `Interval(min,max)`, `Set(v1, v2, …)`, `GreaterThan(x)`

```
H = Parameter("Height", 1, LINEAR, GreaterThan(0));
HandleName = Parameter("Handle", "Plug", Set(FolderList("..\\..\\Handles\\",
"*.tcw")));
```

2D Building Blocks

- **Point helpers**: `PointX(P)`, `PointY(P)`; math: `sin()` `cos()` `tan()` `atan()`
- **Polyline** with corners, fillets & arcs

```
p = Polyline(
  Point(0,0),
  Point(L,0), Fillet(0.5),      // fillet next corner
```

```
  Point(L,H), Arc1(cx,cy),      // arc via center (cx,cy)
  Point(0,H), Point(0,0)
);
```

Text

```
s = Text("Hello", TextFont(0,1,0, "Arial"), TextStyle(CENTER, MIDDLE,
ITALIC));
```

3D Primitives & Solids

```
b = Box(0,0,0, L, W, H);
s = Sphere(R);
c = Circle(R/3);
board = Thickness(Rectangle(W,H), 0.75); // profile → solid

// Booleans
hollow = BooleanSubtract(s, Move(Thickness(c, R*2), 0,0,-R));

// Shell, Fillet, Chamfer, Offset (3D)
s1 = G3Shell(b, Point(L/2, W/2, H), 0.2);
s2 = G3Fillet(s1, Array(Point(L/2,0,0), Point(0,W/2,0)), Array(0.25, 0.25));
s3 = G3Chamfer(s2, Point(L/2, W, H), Array(0.1, 0.2));
s4 = G3Offset(s3, Point(L, W/2, H/2), 0.1);
```

Transforms & Replication

```
// Move / Rotate (about optional center)
T0 = RotateZ(p, A, 0, 0);      // rotate in 2D around (0,0)
T1 = Move(T0, 10, 0, 0);       // translate

// Linear array via Move's copy count
legs = Move(board, 0, 0, (H-0.75)/(K+1), K);  // K copies

// Other axes
Rx = RotateX(obj, 90);
Ry = RotateY(obj, 180);
```

Materials & Properties

```
wood = SetProperties(obj,
  "Material" = "Wood\\Pine",
  "PLANK_Material" = "WOOD-3/4",
  "PenColor" = 0xff,
  "Brush" = "SOLID",
  "BrushStyle" = "SOLID");
```

Input / Output / RefPoint

```
Input(P0, P1, L, A);          // controls UI order & visibility
rp = RefPoint(0, 0, 0);       // insertion handle
Output(obj1, obj2);
Output(rp);                   // separate call is fine
```

Conditionals & Utilities

```
Type = Parameter("Nut Type", "M6", Set("M4","M5","M6"));
res = IF(Type=="M6", NutBase(10,5,1.0,6,10), 0);
minV = min(2,5,1,7,9);  maxV = max(Array(A,B,C));
```

Files, Symbols & Custom Functions

- **Folder listing** for pick-lists: `FolderList(path, mask)`
- **Relative paths** are rooted at a sibling `Macro` folder; use `..\..\..\Drawings` style paths.
- **Static symbols**: `StaticSymbol("..\\..\\Drawings\\Part.tcw")`
- **Call another PPM as a function**: place it in `Macro` and call by its filename (no extension), passing parameters in order.

Quick Map — What Each Confirmed Example Demonstrates

- **SoapDish.ppm** — `Box`, `G3Fillet`, `G3Shell`, `Array`, `Point`, `Output`
- **blend.ppm** — `G3Fillet`, `G3Chamfer`, `G3Offset` chained on a `Box`
- **bend.ppm** — `Rectangle` → `Thickness` → `G3Bend`
- **brush.ppm** — `ParameterPoint`, `Polyline`, `SetProperties(… "Brush" …)`
- **arrow.ppm** — `Input`, `ParameterPoint`, angle math, `Polyline`, `RotateZ`, `Move`, `BrushStyle`
- **text.ppm** — `Text`, `TextFont`, `TextStyle`, `SetProperties("PenColor")`
- **sphere_w_hole.ppm** — `Sphere`, `Circle→Thickness→Move`, `BooleanSubtract`, `SetProperties`, `RefPoint`
- **Table.ppm / Cabinet_T.ppm / cabinet_w(.ppm)** — cabinetry workflow: `Polyline`, `Thickness`, `Move` with copy count, `RotateX/Z`, materials, multiple `Output`, `RefPoint`, `FolderList`
- **Nut.ppm** — enumerated size with `Set(…) + IF` dispatch to `NutBase(…)`
- **Steel Shape.ppm / I.ppm** — polyline profiles with `Fillet(R)` tokens, constrained parameters
- **Flanged Fitting (Elbow / Tee / Cross / 45Lateral).ppm** — 2D layout with `PointX/PointY`, `Arc1`, `Fillet`, `RotateX/Y/Z`, offsetting via trig, grouped `Output`

- **plnp.ppm** — units, typed parameters with `Set/Interval`, boolean subtract, material assignment

Best Practices

- Always declare **Units** and meaningful **parameter constraints**.
- Use `Input(...)` to control UI order and visibility.
- Output a `RefPoint` for predictable insertion.
- Keep identifiers **unique**; avoid circular references.
- Prefer **arrays** for multi-edge fillets/chamfers.
- Use `SetProperties` for materials/colors; keep strings exactly as in your material library.

Troubleshooting Tips

- If edges don't fillet: verify you're passing a **mid-edge Point** (or `Array(Point, …)`) into `G3Fillet/G3Chamfer`.
- If a symbol path fails: remember paths are relative to a sibling **Macro** folder.
- No output? Ensure each Output'd name is **defined once** and that you have **at least one** `Output(...)` call.

This sheet is distilled from working PPMs; copy/paste any snippet into a new `.ppm` file and test in TurboCAD's Parametric Part Script Editor.