Desk Pricing & Risk Mini-Toolkit (Python)

Goal:

Built a small desk-style tool to automate **daily pricing + risk reporting** for a toy options book, with **controls and exception reporting** similar to SOD workflows. This toolkit was extended to incorporate **validation and sanity checks** (input completeness, coherence, and Greek checks) to reflect the **SOD controls and risk governance** expected in the role.

What it does

- **Prices vanilla European options (Call/Put)** using Black–Scholes.
- Computes **Greeks**: **Delta, Gamma, Vega** (analytic).
- Runs **scenario shocks** and computes **scenario PnL**:
    - Spot **+1%**
    - Vol **+1 vol point(0.01)**
- Generates a daily-style **report CSV** + a separate **exceptions/validation CSV**.

Note: Vega convention**:** reported as change in price per **1.00** unit change in vol. For **1 vol point** (0.01) moves, use **Vega_1vp = Vega × 0.01**. Scenario vol_up uses **+0.01**.

Controls / validation checks (SOD-style):

- **Input validation / completeness checks** (e.g., missing or non-positive vol, non-positive time-to-expiry, invalid option type).
- **Model coherence check:** Put–call parity residual (should be ~0 for consistent pricing inputs).
- **Risk sanity check:** Analytic Greeks compared against **finite-difference** Greeks (differences should be small).

Outputs:

- **outputs/report.csv**
  Pricing + Greeks + scenario prices/PnL for valid instruments.
- **outputs/validation_report.csv**
  Exceptions list for instruments failing validation checks (SOD-style breaks).
- **outputs/report.png**
  Quick view of key risk numbers for review