

# Tom Pollak

**Location** York, UK  
**Email** tompollak1000@gmail.com

**GitHub**  
**Website** github.com/tom-pollak  
**Phone** tompollak.me  
(+44) 77400 054268

---

## EDUCATION

**University of York** 2023  
*BEng. Computer Science*  
• Weighted average: 77%, Expected First

**Lady Manners School** 2020  
*A-Level*  
• Further Maths (A), Maths (A), Computer Science (A), Physics (A)

---

## EXPERIENCE

**Cisco Meraki** June - August 2022  
*Full Stack Engineer Intern* London, UK  
• test

---

## PROJECTS

**Automated Horse Betting Program** December 2020 - July 2021  
<https://github.com/tom-pollak/each-way-matcher>

- Discovers undervalued horses by the bookmaker in each-way betting
- Uses an adapted Kelly Criterion strategy to calculate the optimal stake of these undervalued horses
- Uses Python, Pandas and Selenium to scrape the horse races, interacts with Betfair API to place bets
- Runs headless on a Raspberry Pi as a scheduled cron job every day

**Pirate Game** January - February 2022  
<https://github.com/tom-pollak/pirates>

- Pirate video game using Java and LibGDX

**Poker Web Application** April 2019 - July 2020  
<https://github.com/tom-pollak/web-poker>

- Free live poker web app using Python and Django
- Users can create accounts and tables, play poker, and chat with other players
- Uses Django Channels for real-time communication with the users
- Deployed with Docker and Heroku.

**SANS Institute** August 2020  
*FOR500 Windows Forensic Analysis*  
<https://www.sans.org/cyber-security-courses/windows-forensic-analysis>

- Sponsored to take part through my success in the Cyber Discovery programme

**Cyber Discovery** September 2018 - July 2019

- Independently completed the Cyber Discovery programme, run by HM government
- Selected as one of the top 500 (of 28,000) students to attend the Cyber Discovery Elite event in London

---

## SKILLS

**Languages** Python, Java, Haskell, SQL, HTML  
**Tools** Linux, Vim, Git, LaTeX, RegEx, SQLite  
**Technologies** Django, Numpy, Pandas, Selenium, LibGDX, Docker