**Lab - Research Portfolio Requirements**

**Objectives**

In this lab, you will explore Data Analyst Portfolios and create checklist to showcase data analyst skills.

**Part 1: Research Data Analyst Portfolios**

**Part 2: Create Data Analyst Skills Checklist**

**Background / Scenario**

A data analytics portfolio highlights your work and displays your personal branding and communication skills. A data analysist needs to also have a plan to develop the necessary skills and to perform and present their projects to prospective employers. Use this lab as a starting point to begin your Data Analyst Portfolio. As you progress through the course, you can use the activities within the course to build out your first analytic project.

**Required Resources**

* Mobile device or PC/laptop with a browser and an internet connection

**Instructions**

**Part 1: Research Data Analyst Portfolios**

**Step 1: Search for data analyst portfolios.**

1. Using a web browser, search for **example data science portfolios**.

* Example of what a data science portfolio might contain:

Resume/CV: A detailed resume or curriculum vitae highlighting your education, work experience, technical skills, and any certifications relevant to data science.

Projects: This is the heart of your portfolio. Include a selection of data science projects you've worked on. Each project should include:

Project Title: A descriptive title for your project.

Overview: A brief description of the project, its goals, and its significance.

Tools and Technologies: List the tools, programming languages, and libraries you used for the project (e.g., Python, R, TensorFlow, scikit-learn).

Data: Describe the datasets you used, their sources, and any preprocessing steps you performed.

Analysis: Outline the analytical methods, algorithms, and techniques you employed.

Results: Present your findings, insights, and visualizations.

Conclusion and Future Work: Reflect on the project's outcomes, limitations, and potential areas for future improvement or expansion.

GitHub Repository Link: Provide a link to the GitHub repository containing your project code and documentation.

Data Visualization: Showcase your data visualization skills by including visualizations you've created in past projects. Explain the insights derived from these visualizations.

Blog Posts or Articles: If you've written blog posts or articles related to data science, provide links to them in your portfolio. This demonstrates your ability to communicate complex technical concepts effectively.

Data Science Competitions: If you've participated in data science competitions (e.g., Kaggle), mention your achievements, rankings, and any notable solutions you've developed.

Skills and Expertise: List your technical skills, programming languages, statistical methods, machine learning algorithms, and any domain-specific knowledge relevant to data science.

Education and Certifications: Detail your educational background, including degrees earned, relevant coursework, and any certifications you've obtained in data science or related fields.

Testimonials or Recommendations: If possible, include testimonials or recommendations from colleagues, supervisors, or clients who can vouch for your skills and expertise in data science.

Contact Information: Provide your contact information (e.g., email address, LinkedIn profile) so that potential employers or collaborators can reach out to you.

1. Select some portfolios and read through and note the range of content and presentation styles.  
   The website <https://careerfoundry.com/en/blog/data-analytics/data-analytics-portfolio-examples/> examines portfolio examples.

* Content Variety:

Some portfolios may focus heavily on showcasing technical skills and projects, providing detailed descriptions of methodologies, algorithms used, and results achieved.

Presentation Styles:

Minimalistic: Portfolios may adopt a minimalist design with clean layouts, simple color schemes, and concise text to emphasize the content and projects.

**Step 2: List possible projects.**

Make a list of some of the interesting projects that are highlighted in the portfolios that you viewed.

* I choose Naledi demonstrates She achieves this brilliantly with a combination of personal statements and supporting projects. In addition to her portfolio, she also maintains a blog where she writes about her interests. Combined, these aspects all tell us that she believes in the power of data analytics to change the world. That would certainly make us want to hire her!

Also include projects that you might be interested in pursuing, ones where you where would like to conduct data research, such as hobbies, sports, or other interests.

* Including projects related to personal interests, hobbies, or other non-professional areas can demonstrate not only your technical skills but also your passion and creativity in applying data science to diverse domains.

**Step 3: Consider possible portfolio styles.**

Consider how you might want to present your projects in your portfolio. Select a style that reflects your personality and goals.



Examine the examples you have found to determine the media and layout styles that represent you to a prospective employer’s expectations.

**Step 4: Create a portfolio checklist.**

1. Develop a checklist of the components that you see in the portfolios you viewed.

* Want to know if UI design is right for you?

Find out with a free 5-day email short course.

short course modal list itemLearn how the most successful companies design beautiful products.

short course modal list item Discover all the different steps of the UI design process

short course modal list item Find out if user interface design is the right option for you.

1. As you work through the activities in the course, update the checklist as you complete each item on the list.

**Part 2: Create Data Analyst Skills Checklist**

**Step 1: Search for data analyst job skills.**

Search the internet for data analyst jobs in your specific country or region.

One site to consider is <https://www.ziprecruiter.com/n/Data-Analyst-Jobs-Near-Me>.

Sites such as <https://www.thinkful.com/blog/entry-level-data-analyst/> are also useful.

Note the job requirements and essential skills for entry-level or junior data analysts. Some skills to look for include:

* SQL
* Microsoft Excel
* Statistics
* Data visualization
* Presentation and communication skills
* Critical thinking

It is important to separate how these skills relate to entry-level roles and the expectation for more experienced roles.

**Step 2: Create a checklist of data analyst skills.**

Create a checklist of the job requirements for the role of data analyst. As you learn and practice each skill note your level of proficiency – perhaps use a scale like none, basic, intermediate, and advanced.

Continuously update your resume and portfolio as you progress through your checklist.

Analytical Skills:

* Understanding of statistical concepts and methods
* Proficiency Level: [Basic / Intermediate / Advanced]
* Ability to interpret data and draw meaningful insights
* Proficiency Level: [Basic / Intermediate / Advanced]
* Critical thinking and problem-solving skills
* Proficiency Level: [Basic / Intermediate / Advanced]

Technical Skills:

* Proficiency in SQL for data querying and manipulation
* Proficiency Level: [Basic / Intermediate / Advanced]
* Experience with data visualization tools (e.g., Tableau, Power BI)
* Proficiency Level: [Basic / Intermediate / Advanced]
* Familiarity with programming languages such as Python or R for data analysis
* Proficiency Level: [Basic / Intermediate / Advanced]
* Knowledge of data cleaning and preprocessing techniques
* Proficiency Level: [Basic / Intermediate / Advanced]

Data Management:

* Ability to gather, organize, and clean data from multiple sources
* Proficiency Level: [Basic / Intermediate / Advanced]
* Understanding of database concepts and data warehousing
* Proficiency Level: [Basic / Intermediate / Advanced]
* Experience with ETL (Extract, Transform, Load) processes
* Proficiency Level: [Basic / Intermediate / Advanced]

Communication Skills:

* Strong verbal and written communication skills
* Proficiency Level: [Basic / Intermediate / Advanced]
* Ability to present complex data insights to non-technical stakeholders
* Proficiency Level: [Basic / Intermediate / Advanced]
* Experience in creating clear and concise reports and dashboards
* Proficiency Level: [Basic / Intermediate / Advanced]

Domain Knowledge:

* Familiarity with the industry or domain in which the data analyst will be working (e.g., finance, healthcare, marketing)
* Proficiency Level: [Basic / Intermediate / Advanced]
* Understanding of business processes and key performance indicators (KPIs)
* Proficiency Level: [Basic / Intermediate / Advanced]
* Ability to translate business requirements into data analysis tasks
* Proficiency Level: [Basic / Intermediate / Advanced]

Teamwork and Collaboration:

* Ability to work effectively in a team environment
* Proficiency Level: [Basic / Intermediate / Advanced]
* Experience collaborating with cross-functional teams (e.g., data scientists, business analysts, developers)
* Proficiency Level: [Basic / Intermediate / Advanced]
* Strong interpersonal skills and willingness to share knowledge and expertise
* Proficiency Level: [Basic / Intermediate / Advanced]

Continuous Learning:

* Willingness to learn new tools, techniques, and technologies in the field of data analysis
* Proficiency Level: [Basic / Intermediate / Advanced]
* Proactive attitude towards professional development and staying updated with industry trends
* Proficiency Level: [Basic / Intermediate / Advanced

**Part 3: Setting Up a GitHub Repository for Your Portfolio**

GitHub is a hosting platform that provides collaboration and version control service to individuals and organizations. In this part, you will create a personal GitHub account and a repository for your use.

**Step 1: Explore GitHub.**

1. Navigate to [GitHub Documentation](https://docs.github.com/) or search for **github documentation** using a web browser.
2. Click **Get Started** to review some of the basic features of GitHub.

What types of accounts (products) are available on GitHub?

Answer Area

Types and products available on GitHub:

User Accounts:

Free Accounts: Basic GitHub accounts available for individuals and small teams. They offer essential features like public repositories, collaboration tools, and basic issue tracking.

Pro Accounts: Paid accounts offering additional features such as private repositories, advanced collaboration tools, and code review features. Ideal for individual developers and small teams requiring privacy and more advanced project management capabilities.

Team Accounts: Designed for larger teams and organizations, offering enhanced collaboration features, access controls, and centralized administration capabilities. Team accounts provide a scalable solution for managing multiple users and projects within an organization.

Enterprise Accounts:

GitHub Enterprise Cloud: A fully managed, cloud-based solution tailored for enterprises. It offers advanced security features, compliance controls, and scalable infrastructure for large organizations.

GitHub Enterprise Server: A self-hosted version of GitHub Enterprise, allowing organizations to deploy GitHub on their own infrastructure for increased control and customization.

Products and Services:

GitHub.com: The primary web-based platform for hosting repositories, collaborating on code, and managing projects using Git version control.

GitHub Desktop: A desktop application that provides a graphical user interface for working with Git repositories. It simplifies common Git workflows and is available for Windows and macOS.

GitHub CLI (Command Line Interface): A command-line tool that allows users to interact with GitHub repositories and perform common tasks directly from the terminal.

GitHub Actions: A CI/CD (Continuous Integration/Continuous Deployment) service provided by GitHub. It allows users to automate workflows, build/test code, and deploy applications directly from their GitHub repositories.

GitHub Packages: A package hosting service integrated with GitHub repositories. It allows users to publish and share software packages such as npm packages, Docker containers, and Maven artifacts.

GitHub Gist: A service for sharing code snippets, text, and files quickly. Gists can be public or private and are often used for sharing examples, documentation, or short code snippets.

GitHub Pages: A static site hosting service provided by GitHub. It allows users to publish websites directly from their GitHub repositories, making it easy to create and host personal blogs, project documentation, or portfolio websites.

Show Answer

What can you do with a personal account?

Answer Area

Here are some of the key capabilities and features available to users with a personal GitHub account:

Host Public Repositories: You can create public repositories to host your code and make it accessible to the public. Public repositories are freely accessible and can be viewed, forked, and contributed to by other GitHub users.

Collaborate with Others: You can collaborate with other developers by inviting them to contribute to your repositories or by contributing to other users' repositories through pull requests. GitHub provides tools for code review, issue tracking, and discussion, facilitating collaboration among team members.

Contribute to Open Source Projects: With a personal GitHub account, you can contribute to open source projects hosted on GitHub by submitting pull requests, reporting issues, or participating in discussions. Contributing to open source projects is a great way to gain experience, build your reputation, and give back to the community.

Manage Projects: GitHub provides project management features such as project boards, milestones, and labels to help you organize and track the progress of your projects. You can create and manage projects directly from your repositories, making it easy to track tasks, prioritize work, and coordinate with team members.

Discover and Explore Code: You can explore millions of public repositories on GitHub to discover interesting projects, learn new technologies, and find code examples and resources. GitHub's search functionality allows you to search for repositories, users, topics, and more based on specific criteria.

Showcase Your Work: GitHub serves as a platform for showcasing your coding projects, portfolio, and contributions to the developer community. You can customize your profile, highlight your skills and expertise, and share links to your repositories, personal website, and social media profiles.

Access Learning Resources: GitHub offers a wealth of learning resources, tutorials, and guides to help you improve your coding skills, learn new technologies, and stay updated with the latest trends in software development. You can explore GitHub Learning Lab, GitHub Guides, and other educational resources to enhance your knowledge and proficiency.

Show Answer

**Step 2: Create a personal account.**

A GitHub account is necessary to create a repository. A repository is where you can organize your project and store your files and folders for a single project.

In this step, you will create a personal GitHub account.

1. Navigate to [GitHub](https://github.com/) to sign up for an account.  
   **Note**: If you already have a GitHub account, verify that you have already logged out of the account.
2. In the **Email address** field, enter a valid email address and click **Sign up for GitHub** to continue to the Welcome screen.
3. In the Welcome screen, verify your email address, create a password, enter a username, answer the question regarding receiving products and updates via email, and solve puzzles. Click **Create account** to create the new user account. You will receive a code from GitHub in the provided email address.
4. Enter the provided launch code and your account is ready for use.
5. Answer personal prompts and click **Continue**. Click **Continue** on the feature screen without any selections. Click **Continue free** after reviewing the features of the free account.

**Step 3: Create a new private repository.**

Now that you have a new user account, you will create a new private repository.

1. Log into GitHub using your new user account as necessary.
2. Click **Create a new repository**.
3. Enter a repository name of your choice in the **Repository name** field. Provide a description of the repository if desired. Select **Private** to a create a private repository. Select **Add a README file**. Click **Create repository** to continue.

**Step 4: Manage files in the repository.**

Files can be uploaded to the repository or create new files with the repository. If you have already created a README file, you can copy the content of the README file into the existing README file in the repository.

In this step, you will upload a file to the repository. For more file management information, refer to the GitHub docs regarding repositories. (<https://docs.github.com/en/repositories>).

1. In the newly created repository, click**Add file** > **Upload files** to upload a new file under the <> Code tab.
2. You can either drag or click **choose your files** to add the file to the repository. Click **Commit changes** to save the changes directly to the main branch.

**Reflection Questions**

Reflect upon the skills required by a data analyst role and consider what additional, or enhanced, skills that are required to develop the projects for your portfolio that interest you and where you see your data analyst career leading you.

**Challenge Activities**

The Square Kilometer Array (SKA) telescope project (<https://www.skatelescope.org/the-ska-project/>) and the James Webb Telescope (<https://www.nasa.gov/mission_pages/webb/main/index.html>) are very long-term astronomy projects that will produce almost unimaginable volumes of data over the lifetime of each project.

Think about the data skills that will have to be applied to analyze this data, and what processes may have to be developed to manage this analysis into the future.

Show All AnswersClear All Responses