**AWS S3 Class Work: Problem Statement**

You are tasked with configuring and managing an S3 bucket in AWS for a fictional e-commerce platform. The platform requires scalable, secure, and cost-effective storage for its website's static files, including images, HTML pages, and backup data. As part of the setup, you need to ensure the following features:

**Problem Statement**

1. **Bucket Creation and Management**:
   * Create an S3 bucket with appropriate properties for storing website assets.
   * Ensure the bucket has the necessary configurations to support a public-facing static website and object versioning.
2. **Object Management**:
   * Upload various types of files (HTML, CSS, images, and backups) to the S3 bucket.
   * Ensure the files have the correct metadata and are securely accessible according to platform requirements.
3. **Versioning and Recovery**:
   * Enable versioning for objects in the bucket to ensure that all versions of uploaded files are maintained.
   * Verify that previous versions of files are retained and can be recovered if needed.
4. **Bucket Security and Access**:
   * Implement an appropriate bucket policy that grants public access to website files but restricts unauthorized access to other objects.
   * Ensure that the bucket adheres to security best practices while allowing the necessary permissions for public objects.
5. **Cost Management**:
   * Implement a lifecycle policy to transition infrequently accessed data (such as backups) to a more cost-effective storage class (e.g., Glacier) after a specified number of days.
   * Ensure the lifecycle policy balances cost management and data retrieval needs.
6. **Static Website Hosting**:
   * Configure the bucket to host a static website.
   * Ensure the website is publicly accessible, and users can access an index.html file as the homepage.
   * Implement a custom error page that users are redirected to in case of incorrect URLs or unavailable resources.

**Requirements to Fulfill**

* **Bucket Properties**:
  + Enable versioning on the bucket.
  + Set appropriate permissions for static website hosting.
* **Upload and Manage Objects**:
  + Upload various objects (e.g., HTML, images) with appropriate metadata.
  + Ensure object URLs are publicly accessible.
* **Security Policies**:
  + Create a bucket policy that provides secure public access to static assets.
  + Ensure access control for private objects.
* **Lifecycle Management**:
  + Create a lifecycle policy to manage cost efficiency by transitioning objects (e.g., older backups) to cheaper storage tiers after a defined period.

**Deliverables**

1. **Bucket Name**: Provide the name and region of the bucket created.
2. **Bucket Policy**: Submit a screenshot or JSON output of the bucket policy used for public access.
3. **Object URLs**: Provide the public URLs of at least two different objects (e.g., an HTML page and an image).
4. **Versioning**: Submit evidence (screenshot) that versioning is enabled, and show that an object version has been updated.
5. **Lifecycle Policy**: Submit a screenshot or configuration details of the lifecycle policy for transitioning objects to a different storage class.
6. **Static Website URL**: Submit the public URL of the static website hosted on S3.