Lecture 08: Refactoring – Using Persist Method

Updated Code with Persist Method

```java
public class Main {
    public static void main(String[] args) {
        Student s1 = new Student();
        s1.setName("Gaurav");
        s1.setRollNo(105);
        s1.setAge(22);

        Configuration cfg = new Configuration();
        cfg.addAnnotatedClass(com.telusko.Student.class);
        cfg.configure();

        SessionFactory sf = cfg.buildSessionFactory();
        Session session = sf.openSession();

        Transaction transaction = session.beginTransaction();

        session.persist(s1);  // Using persist instead of save

        transaction.commit();

        session.close();
        sf.close();

        System.out.println(s1);
    }
}
```

Persist vs Save Method

- **persist()**: JPA standard method, preferred for new code
- **save()**: Hibernate-specific, deprecated in newer versions
- **Functionality**: Both methods save transient objects to the database.
- **Return value**: save() returns generated ID; persist() doesn't

Best Practices

1. Always use transactions for data modifications
2. Close session and session factory properly
3. Use persist() for better portability
4. Handle exceptions appropriately