BOOK CLASS:

```
package bookmapping;

import java.util.HashSet;
import java.util.Set;
import javax.persistence.Column;
import javax.persistence.DiscriminatorColumn;
import javax.persistence.DiscriminatorType;
import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.Inheritance;
import javax.persistence.InheritanceType;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToMany;
import javax.persistence.Table;

@Entity
@Table(name="Book_Details")
@Inheritance(strategy=InheritanceType.TABLE_PER_CLASS)
@DiscriminatorColumn(name="book_type",discriminatorType=DiscriminatorType.STRING)
@DiscriminatorValue("book")
public class Book {
		@ManyToMany
		@JoinTable(name="Book_Author_Relation",
			joinColumns=@JoinColumn(name="book_id"),
			inverseJoinColumns=@JoinColumn(name="author_name"))
		private Set<Author> author = new HashSet<Author>();



		public Set<Author> getAuthor() {
			return author;
		}

		public void setAuthor(Set<Author> author)
			{ this.author = author;
		}


	@Column(name="book_id")
	@GeneratedValue
	@Id
	int id;
	@Column(name="book_title")
	String title;

	@Column(name="book_price")
	double price;
	@Column(name="isbn_no")
	String isbnno;
	@Column(name="publisher")
	String publisher;
```

```
	@Column(name="edition")
	String edition;
	public Book() {
		super();
		// TODO Auto-generated constructor stub
	}
	public int getId() {
		return id;
	}
	public void setId(int id) {
		this.id = id;
	}
	public String getTitle() {
		return title;
	}
	public void setTitle(String title) {
		this.title = title;
	}
	public double getPrice() {
		return price;
	}
	public void setPrice(double price) {
		this.price = price;
	}
	public String getIsbnno() {
		return isbnno;
	}
	public void setIsbnno(String isbnno) {
		this.isbnno = isbnno;
	}
	public String getPublisher() {
		return publisher;
	}
	public void setPublisher(String publisher) {
		this.publisher = publisher;
	}
	public String getEdition() {
		return edition;
	}
	public void setEdition(String edition) {
		this.edition = edition;
	}



}
```

EBOOK CLASS:

```
package bookmapping;

import javax.persistence.Column;
import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;
```

```
import javax.persistence.Table;
@Entity
@Table(name="EBook_Details")
@DiscriminatorValue("ebook")

public class Ebook extends Book {
	@Column(name="download_url")

	String downloadurl;
	@Column(name="size_in_mb")
	int size_in_mb;
	public Ebook() {
		super();
		// TODO Auto-generated constructor stub
	}
	public String getDownloadurl() {
		return downloadurl;
	}
	public void setDownloadurl(String downloadurl) {
		this.downloadurl = downloadurl;
	}
	public int getSize_in_mb() {
		return size_in_mb;
	}
	public void setSize_in_mb(int size_in_mb) {
		this.size_in_mb = size_in_mb;
	}



}
```

PAPERBOOK CLASS:

```
package bookmapping;

import javax.persistence.Column;
import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;
import javax.persistence.Table;

@Entity
@Table(name="paperbook_Details")
@DiscriminatorValue("paperbook")
public class PaperBook extends Book {
@Column(name="shipping_weight")
	float shipping_weight;
@Column(name="instock")
	int instock;
	public PaperBook() {
		super();
		// TODO Auto-generated constructor stub
	}
	public float getShipping_weight() {
```

```
		return shipping_weight;
	}
	public void setShipping_weight(float shipping_weight) {
		this.shipping_weight = shipping_weight;
	}
	public int getInstock() {
		return instock;
	}
	public void setInstock(int instock) {
		this.instock = instock;
	}


}
```

AUTHOR CLASS:

```
package bookmapping;

import java.util.HashSet;
import java.util.Set;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToMany;
import javax.persistence.Table;

@Entity
@Table(name="Author")
public class Author {

	@ManyToMany
	@JoinTable(name = "Author_Book_Relation", joinColumns =
	@JoinColumn(name = "author_name"), inverseJoinColumns = @JoinColumn(name =
	"book_id"))
	private Set<Book> book = new HashSet<Book>();

	public Set<Book> getBook() {
		return book;
	}

	public void setBook(Set<Book> book) {
		this.book = book;
	}

	@Column(name="author_id")
	@Id
	int id;
	@Column(name="author_name")
	String name;
	@Column(name="email_id")
	String email;
	@Column(name="gender")
	String gender;
```

```
	@Column(name="url")
	String url;
	public Author() {
		super();
		// TODO Auto-generated constructor stub
	}
	public int getId() {
		return id;
	}
	public void setId(int id) {
		this.id = id;
	}
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public String getEmail() {
		return email;
	}
	public void setEmail(String email) {
		this.email = email;
	}
	public String getGender() {
		return gender;
	}
	public void setGender(String gender) {
		this.gender = gender;
	}
	public String getUrl() {
		return url;
	}
	public void setUrl(String url) {
		this.url = url;
	}



}
```

SOLUTION (MAIN CLASS):

```
package bookmapping;


import java.io.BufferedInputStream;

import java.io.BufferedReader;
```

```
import java.io.IOException;
import java.io.InputStreamReader;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class Solution {

	public static void main(String[] args) throws NumberFormatException, IOException {
		// TODO Auto-generated method stub
		SessionFactory sf=new Configuration().configure().buildSessionFactory();
	Session session=sf.openSession();
	session.beginTransaction();
	BufferedReader b=new BufferedReader(new InputStreamReader(System.in));


	Book book=new Book();
	book.setId(1);
	book.setTitle("tintin");
	book.setPrice(201.7d);
	book.setIsbnno("1762718");
	book.setPublisher("sa publisher");
	book.setEdition("1st Edition");
	session.save(book);
```

```
Book book1=new Book();
book1.setId(2);
book1.setTitle("two states");
book1.setPrice(655.7d);
book1.setIsbnno("6732372");
book1.setPublisher("na publisher");
book1.setEdition("3rd Edition");
session.save(book1);

Ebook ebook=new Ebook();
ebook.setDownloadurl("oracle_software_downloads");
ebook.setSize_in_mb(256);
session.save(ebook);

Ebook ebook1=new Ebook();
ebook1.setDownloadurl("emc_software_downloads");
ebook1.setSize_in_mb(4528);
session.save(ebook1);

PaperBook paper=new PaperBook();
paper.setShipping_weight(245);
paper.setInstock(3);
session.save(paper);
```

```
        PaperBook paper1=new PaperBook();
        paper1.setShipping_weight(67.8f);
        paper1.setInstock(2);
        session.save(paper1);


        session.getTransaction().commit();
        session.close();
        sf.close();


            }
}
```