

Inheritance:

1. `public class Bishop extends Piece {`
ChessProject\src\game\pieces\Bishop.java:13
2. ChessProject\src\game\pieces\King.java:13
3. ChessProject\src\game\pieces\Knight.java:13
4. ChessProject\src\game\pieces\Pawn.java:13
5. ChessProject\src\game\pieces\Queen.java:14
6. ChessProject\src\game\pieces\Rook.java:13

Polymorphism:

1. `@Override`
`public List<Move> possibleMoves(Board board) {`
ChessProject\src\game\pieces\Bishop.java:23

Encapsulation:

```
private Color color;  
private Position position;  
private boolean firstMove;
```

ChessProject\src\game\pieces\Piece.java:11-13

Abstraction:

```
public abstract class Piece implements Movable {
```

ChessProject\src\game\pieces\Piece.java:9

Upcasting:

```
private final Piece piece;
```

ChessProject\src\game\Move.java:13

Downcasting:

```
if (!(piece instanceof Pawn)) {
```

ChessProject\src\game\Move.java:60

Overloading:

```
public Rook(Position position, Color color) {  
    super(position, color);  
}
```

```
public Rook() {  
    super();  
}
```

ChessProject\src\game\pieces\Rook.java:16-21

Singleton pattern:

```
game = Game.getInstance();
```

ChessProject\src\ui\ChessGUI.java:35