1. Create a class called **Point** that has two attributes x-coordinate and y-coordinate. It should have method called **dist_from_origin** that should give how far point is from origin. [Hint: $d = \sqrt{x^2 + y^2}$]. Operator **+** should add as (x1+x2, y1+y2), **–** should implement (x1-x2, y1-y2). **Print** should display **P(x, y)**. Also, overwrite relational operators that shows if point is smaller or greater based on distance from origin. (i.e. Overwrite <, >, <=, >=, == and !=). Initialize two point objects to demo above methods

2. Create a class called **BankAccount** that has private attribute **balance** (optional: default 0) and public attribute **owner**. It should have method **deposit** that only accepts positive number & adds to user's balance. Add another method **withdraw** that accepts positive number and withdraws from account. After withdraw display the remaining amount or message insufficient balance. Create property attribute **balance** to display it. Create method called **transfer** that transfer balance from one account to another. Initialize two bankaccount to show demo.

3. Create a class called **Employee** that has attribute **name**, **dob**, **age**, **salary**, **skill_sets.** Create class **Developer** inheriting Employee that has additional attributes github_profile, is_fullstack. Developer should have method **display_profile** thaf displays **{name} has skills: s1, s2. Git profile: {profile}** .Create class **HR** inheriting Employee that has attributes **is_manager**, **onboard_rate**. It should also have method **display_profile** that displays **{name} has skills: s1, s2. Onboard rate: {rate}.** 5 developers and 2 HR joined the company. Initialize the object for them. Also display the employee's name who have both Python and Git skill.