

```

public class TestBufferedReader {
    public static void main(String[] args) {
        System.out.print("Entrez le nom du fichier : ");
        Scanner sc = new Scanner(System.in);
        String nomFichier = sc.nextLine();
        BufferedReader bufferedReader = null;

        try {
            System.out.println("Lecture du fichier : " + nomFichier);
            bufferedReader = new BufferedReader(new FileReader(nomFichier));
            System.out.println("*** DEBUT DU FICHER ***");
            String ligne;
            while((ligne = bufferedReader.readLine()) != null){
                System.out.println(ligne);
            }
            System.out.println("*** FIN DU FICHER ***");
            bufferedReader.close();
        } catch (FileNotFoundException e) {
            System.out.println("Erreur : Fichier non trouvé !");
        } catch (IOException e) {
            System.out.println("Erreur : " + e.getMessage());
        }
    }
}

public class ListeDeFichiers {
    public static void main(String[] args) throws IOException {
        File f = new File(args[0]);
        File[] paths;

        paths = f.listFiles();

        for(File path:paths){
            System.out.println("Informations de : " + path.getName());
            if(path.isDirectory()){
                System.out.println("C'est un dossier");
            } else {
                System.out.println("C'est un fichier");
                System.out.println("Taille : " + path.length());
                if(path.canExecute()){
                    System.out.println("Peut être exécuté");
                }
                if(path.canRead()){
                    System.out.println("Peut être lu");
                }
                if(path.canWrite()){
                    System.out.println("Peut être modifié");
                }
            }
            System.out.println(".....");
        }
    }
}

```

```

public class TestEcriture {
    public static void main(String[] args) throws IOException {
        double entree = 10;

        System.out.print("Entrez le nom du fichier à créer : ");
        Scanner sc = new Scanner(System.in);
        String nomFichier = sc.nextLine();
        PrintWriter printWriter = new PrintWriter(new FileWriter(nomFichier));

        System.out.print("Entrez la valeur : ");
        while((entree = sc.nextDouble()) != 0){
            System.out.println(entree + " a pour carree " + Math.pow(entree, 2));
            printWriter.println(entree + " a pour carree " + Math.pow(entree, 2));
            System.out.print("Entrez la valeur : ");
        }

        printWriter.close();
    }
}

public class AffichageMAC {
    public static void main(String args[]) throws Exception {
        URL macvender = new URL("https://api.macvendors.com/");

        for(int i=0; i < args.length; i++){
            URLConnection rc = new URL(macvender + "/" + args[i]).openConnection();
            BufferedReader in = new BufferedReader(new InputStreamReader(rc.getInputStream()));
            System.out.println(in.readLine());
        }
    }
}

```

```

ArrayList<Ville> maListe = new ArrayList<Ville>();
maListe.add(rennes);
maListe.add(bordeaux);
maListe.add(Paris);

System.out.println("Appel de affiche des obj dans ArrayList");
/* Lambda - Pas dans le programme
maListe.forEach((ville) -> {
    System.out.println(ville.getClass());
    ville.affiche();
});
*/
/*System.out.println("Affichage de ma liste avec for classique");
for(int i=0;i<maListe.size();i++){
    maListe.get(i).affiche();
}*/

/*
System.out.println("Affichage de ma liste avec foreach");
for(Ville vil: maListe){
    //vil.affiche();
    //System.out.println(vil);
    if(vil.getClass() == Capitale.class){
        ((Capitale) vil).afficheCap();
    } else {
        vil.afficheVille();
    }
}*/

System.out.println("Affichage de ma liste avec Iterator");
Iterator<Ville> it = maListe.iterator();
while(it.hasNext()){
    System.out.println(it.next());
}

```