Test Case for SignUp Function

|  |  |  |  |
| --- | --- | --- | --- |
| TC Sr. No. | Test Case Scenario | Test Case Method | Status |
| TC1 | Create SingUp object.  SignUp signup = new SignUp( ); | Email {  Main ( ) { }  } | Fail  “No such class available” |
| TC2 | Create SingUp object.  SignUp signup = new SignUp ( ); | SignUp {  public Signup ( ) { }  } | Pass |
| TC3 | Pass user full name, email, password.  SignUp signup = new SignUp( “Vinay” , ”[vinay@gmail.com](mailto:vinay@gmail.com)” , ”123456” ); | SignUp {  public Signup ( ) { }  } | Fail  “No matching constructor found” |
| TC4 | Pass user full name, email, password.  SignUp signup = new SignUp ( “Vinay” , ”[vinay@gmail.com](mailto:vinay@gmail.com)” , ”123456” ); | SignUp{  public Signup ( String fullName, String email, String password ) { }  } | Pass |
| TC5 | Validate fields.  fieldValidation ( ); | SignUp{  public Signup ( String fullName, String email, String password ) { }  } | Fail  “No such function found” |
| TC6 | Validate fields.  fieldValidation ( ); | SignUp{  public Signup ( String fullName, String email, String password ) { }  public boolean fieldValidation ( ) { }  } | Pass |
| TC7 | Validate name.  nameValidation ( ); | public boolean fieldValidation ( ) { } | Fail  “No such function” |
| TC8 | Validate name.  nameValidation ( ); | public boolean fieldValidation(){}  nameValidation ( ) { } | Pass |
| TC9 | Validate name for no characters.  nameValidation ( “” ); | public boolean fieldValidation ( ) { }  nameValidation ( String fullName ) { } | Fail  “Empty name” |
| TC10 | Check name for no characters.  nameValidation ( “Vinay” ); | public boolean fieldValidation ( ) { }  nameValidation ( String fullName ) { } | Pass |
| TC11 | Check name for characters only.  nameValidation ( “Vinay” ); | nameValidation ( String fullName ) {  } | Fail  “Matching criteria not found” |
| TC12 | Check name for characters only.  nameValidation ( “Vinay” ); | nameValidation ( String fullName ) {  If ( isNan (fullName) ) {  return true;  }  } | Pass |
| TC13 | Validate email.  emailValidation ( “Vinay” ); | public boolean fieldValidation ( ) { } | Fail  “No function found for email” |
| TC14 | Validate email for non empty email.  emailValidation ( “vinay” ); | public boolean fieldValidation ( ) { }  emailValidation ( String email ) { } | Pass |
| TC15 | Validate email for @ character.  emailValidation( “vinay” ); | emailValidation ( String email ) {  if( email.contains ( “@” ) )  return true;  else  return false;  } | Fail  “@ is not added” |
| TC16 | Validate email for @ character.  emailValidation ( “vinay@” ); | emailValidation ( String email ) {  if( email.contains(“@”) )  return true;  else  return false;  } | Pass |
| TC17 | Validate email for domain name.  emailValidation ( “vinay@” ); | emailValidation ( String email ) {  If ( email.match(“^\w+@[a-zA-Z\_]+?\.[a-zA-Z]{2,3}$” ) )  return true;  else  return false;  } | Fail  “Domain name is not provided” |
| TC18 | Validate email for domain name.  emailValidation ( “[vinay@gmail.com](mailto:vinay@gmail.com)” ); | emailValidation ( String email ) {  If ( email.match(“^\w+@[a-zA-Z\_]+?\.[a-zA-Z]{2,3}$” ) )  return true;  else  return false; | Pass |
| TC19 | Validate password for empty string.  passwordValidation ( “” ); | passwordValidation ( String password ) {  If ( password.equals(“ ”) ) ) {  return true;  }  else {  return false;  }  } | Fail  “Password is empty” |
| TC20 | Validate password for empty string.  passwordValidation ( “123456” ); | passwordValidation ( String password ) {  If ( password.equals(“ ”) ) ) {  return true;  }  else {  return false;  }  } | Pass |
| TC21 | Validate password for 8 min character with one lower and one upper case alphabet.  passwordValidation ( “123456” ); | boolean passwordValidation ( String password ) {  If ( password.match("^(?=.\*[A-Za-z])(?=.\*\d)[A-Za-z\d]{8,}$" ) ) {  return true;  }  else {  return false;  }  } | Fail  “Password does match for password criteria” |
| TC22 | Validate password for 8 min character with one lower and one upper case alphabet.  passwordValidation ( “123456vY” ); | boolean passwordValidation ( String password ) {  If ( password.match("^(?=.\*[A-Za-z])(?=.\*\d)[A-Za-z\d]{8,}$" ) ) {  return true;  }  else {  return false;  }  } | Pass |
| TC23 | Validating email already exist  ifEmailExist(“[vinay@gmail.com](mailto:vinay@gmail.com)”) | boolean isEmailExist(“[vinay@gmail.com](mailto:vinay@gmail.com)”){    } | Pass  “Email doesn’t exist” |
| TC24 | Validating email confirmation.  confirmEmail ( “[vinay@gmail.com](mailto:vinay@gmail.com)” ); | boolean isEmailReceived ( String email ) {  return false;  } | Fail  “Email not verified” |
| TC25 | Validating email confirmation.  confirmEmail ( “[vinay@gmail.com](mailto:vinay@gmail.com)” ); | boolean isEmailReceived( String email ) {  return true;  } | Pass |