Group size: Two students

Part A: Secure file transfer between Alice (student A) and Bob (student B)
1. Create RSA (2048) key pairs for Alice and Bob and exchange public keys over email. Password protect your respective private keys
2. Alice creates a text file named SA.key with this info <symmetric encryption algo, passphrase and no. of iterations>. Bob also does same thing. These serve like keys for decrypting files exchanged in each way.
3. Alice has to securely send SA.key to Bob. Devise a mechanism in such a way that only Bob can see that message and verify it indeed came from Alice. Similarly, Bob has to securely send his SB.key to Alice and prove its authenticity and integrity.
4. Alice encrypts a large file (some PDF) with SA.key and sends it to Bob so that he could decrypt it with the same SA.key. Similarly, Bob should send some large file securely to Alice.

Part B: Alice (Browser), Bob (web server) and Charlie (Root CA)

1. Charlie generates a self-signed certificate named charlier-ca.pem or charlie-ca.crt as he is the root CA
2. Bob generates CSR named bob-browser.csr and emails it to Charlie for providing end-user cert named bob-browser.crt
3. Alice gets crts of Charlie and Bob and verify that Bob's certificate is valid.

Deliverables: Submit a report (Google Doc) link explaining how you did you complete these two PARTs. Suffix Alice and Bob with last two digits of your rollNos. Include contents of .pem, .crt/.cer, .csr files generated for Alice, Bob and Charlie as part of this tutorial in your report using commands like
openssl x509 -in <cert-name> -text
openssl req -in <csr-name> -text
openssl pkey -in <public-key-name> -text -pubin
cat <encrypted-private-key>
cat SA.key
cat SB.key