1 Task 1: Creating a Secure Code (Hashing and Encoding)

1.1 What is Happening?

This task shows how a secret word or phrase ("jabbarkhan") is turned into a scrambled, secure code. This process is often used to store passwords safely. We mix the original word with a random secret number (called a "salt"), use a high-security scrambler (called a "hash"), and then encode the final result so it can be easily sent or saved.

1.1.1 Workflow Diagram and Configuration

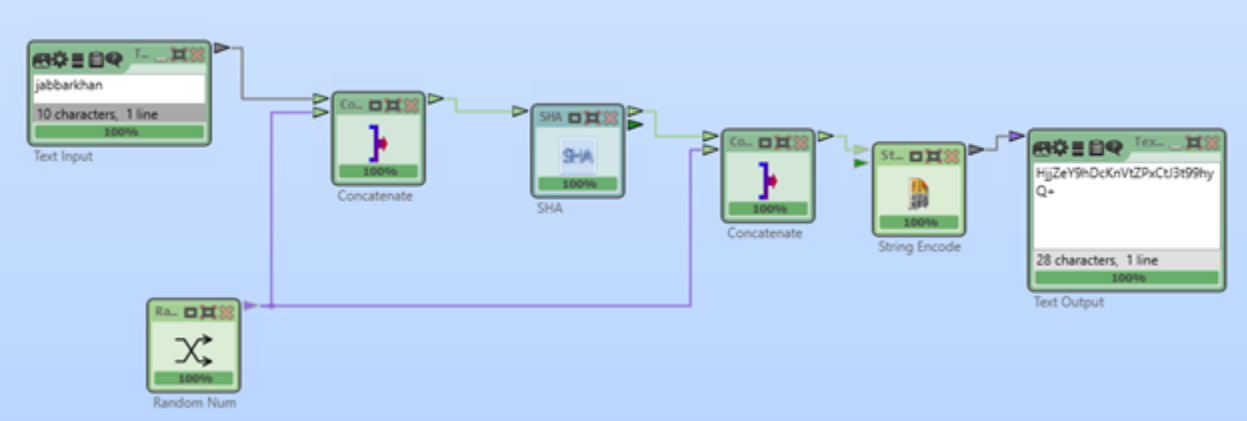


Figure 1: The steps for scrambling and securing the input word.

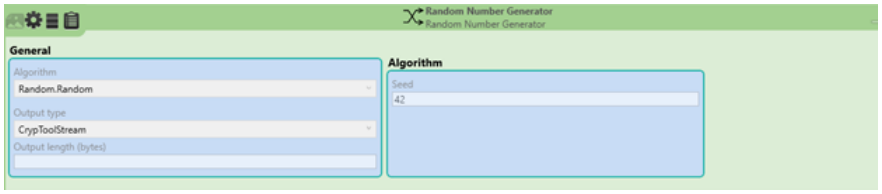


Figure 2: Setting up the Random Secret Number Generator (Seed: 42).

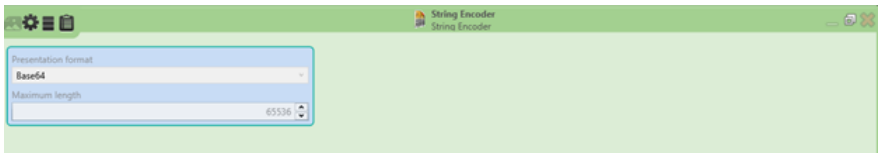


Figure 3: Setting up the Final Code Encoder (Base64 format).

1.1.2 Step-by-Step Process

1. **Input Word**: We start with the secret word: `jabbarkhan`.
2. **Random Secret Number (Salt)**: A temporary, random number is generated using a specific starting point (Seed: 42). This number is the "salt" that makes the final code unique, even if someone uses the same input word.

3. **First Mix**: The input word (`jabbarkhan`) is instantly mixed together with the Random Secret Number.

4. **High-Security Scrambler (SHA)**: The mixed string goes through a strong, one-way scrambler (SHA) that creates a fixed-length output, called a hash. You cannot unscramble a hash back to the original word.

5. **Second Mix**: The scrambled hash is mixed again with the *same* Random Secret Number. This adds an extra layer of protection.

6. **Final Code Encoder (Base64)**: The result is converted into a standard format (`Base64`) using only letters, numbers, and symbols. This makes it easy to handle and transfer.

7. **Output Code**: The final, secure code is `HjZeY9hDcknVlZPzCt3t99HyQ+`.

1.2 Summary of Tools Used

Tool Name	What It Does	Key Setting
Text Input	Provides the starting word.	`jabbarkhan`
Random Num	Creates a secret, random number (salt).	Seed: 42
Concatenate (1 & 2)	Simply joins two pieces of information together.	Mixing data
SHA	Performs the powerful, one-way scrambling (hashing).	N/A
String Encode	Changes the scrambled data into a readable text format.	Format: Base64
Text Output	Shows the final secure code.	`HjZeY9hDcknVlZPzCt3t99HyQ+`

2 Task 2: Electronically Signing a Document (PDF E-Signature)

2.1 What is Happening?

This task shows how to use a software program (Adobe Acrobat Reader) to create and add a personal signature to a digital document (a resume). This is done completely electronically, saving time and paper.

2.1.1 Interface Views

2.1.2 Signature Creation and Application

2.2 Step-by-Step Process

1. **Open Program**: The Adobe Acrobat Reader program is opened, showing the user's files and available quick actions (Figure 4).

2. **Find the Tool**: The user browses the "All tools" menu to find the feature for adding signatures.

3. **Go to E-Sign**: The dedicated "E-Sign" panel is opened, confirming the option to "FILL AND SIGN YOURSELF" (Figure **??**).

4. **Create Signature**: The user clicks "Add signature." They choose the **Type** option and type in their name, `jabbar`, in a nice handwriting style font. This signature is saved for future use (Figure 7).

5. **Place Signature**: The saved signature (`jabbar`) is selected and placed directly onto the resume document, `Jabbar_khan_resume.pdf` (Figure 8).

6. **Final Step**: After adding the signature, the program suggests saving a special "certified copy" which keeps a record of when and how the document was signed.

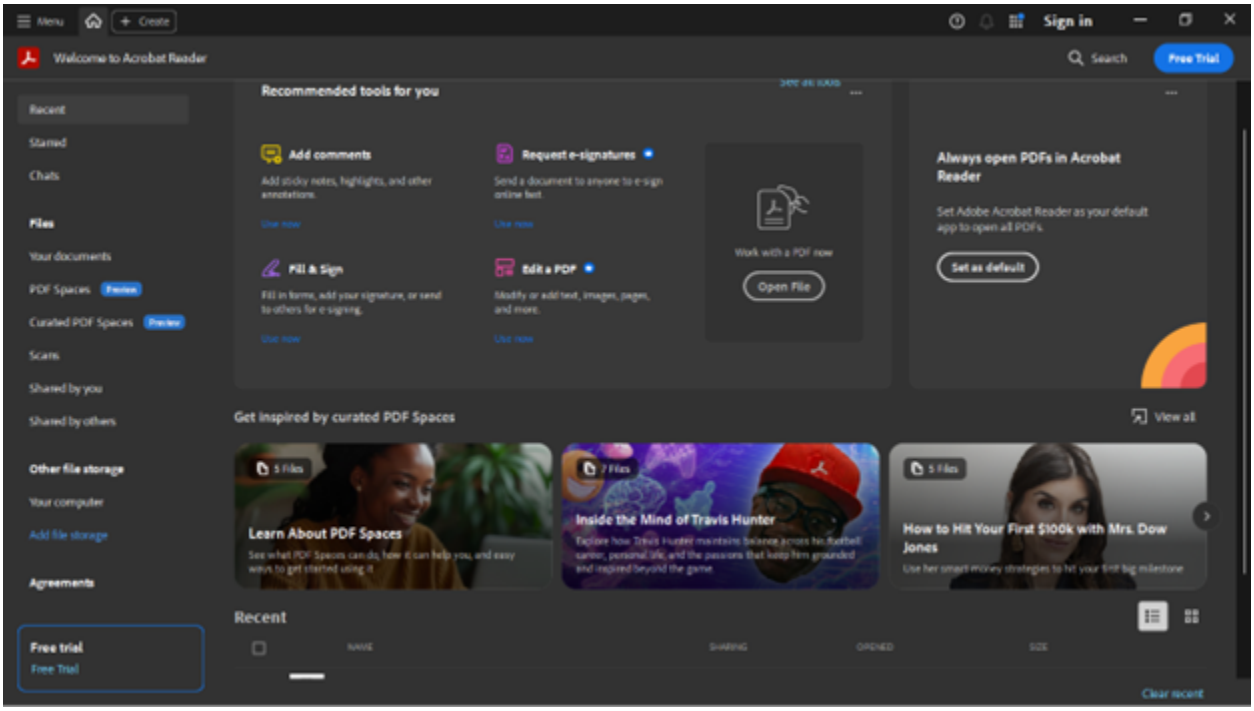
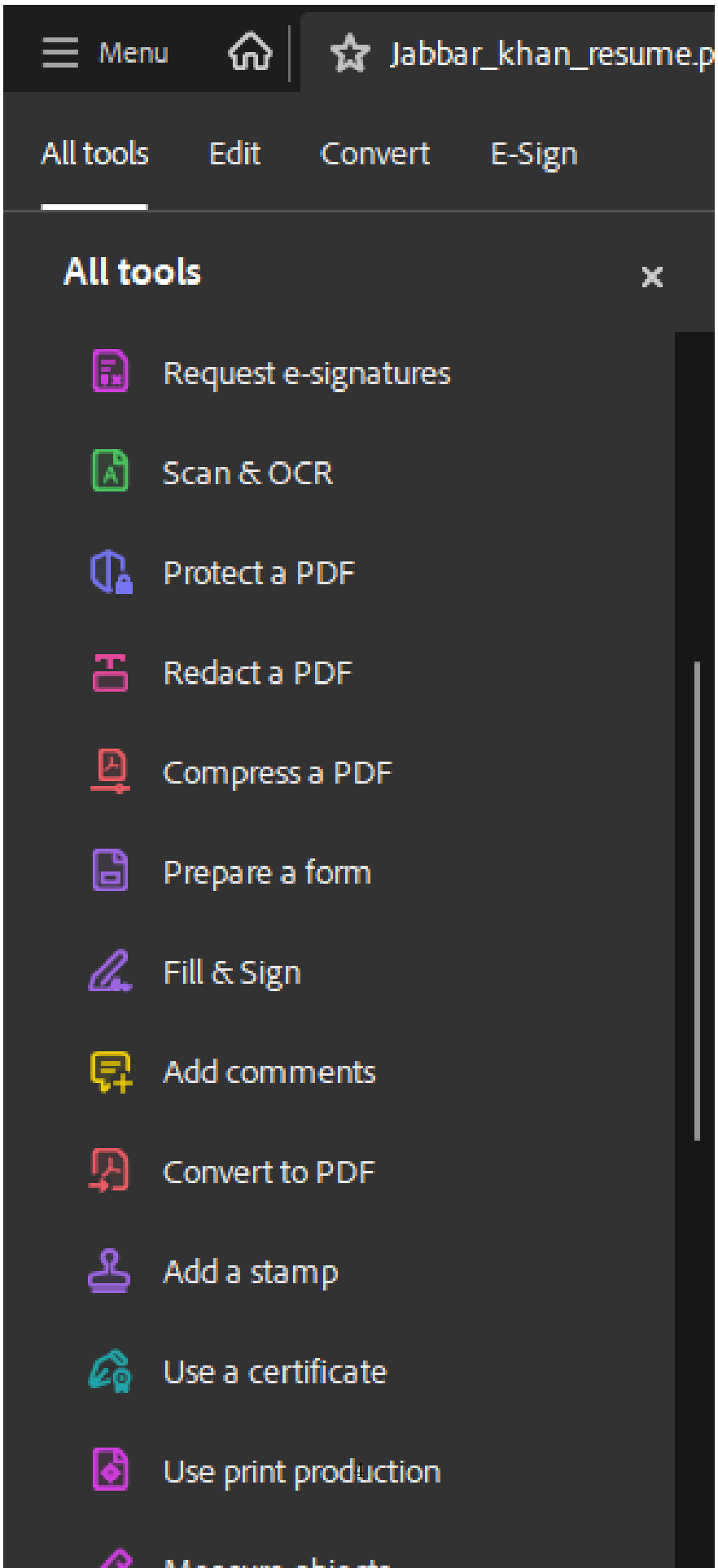
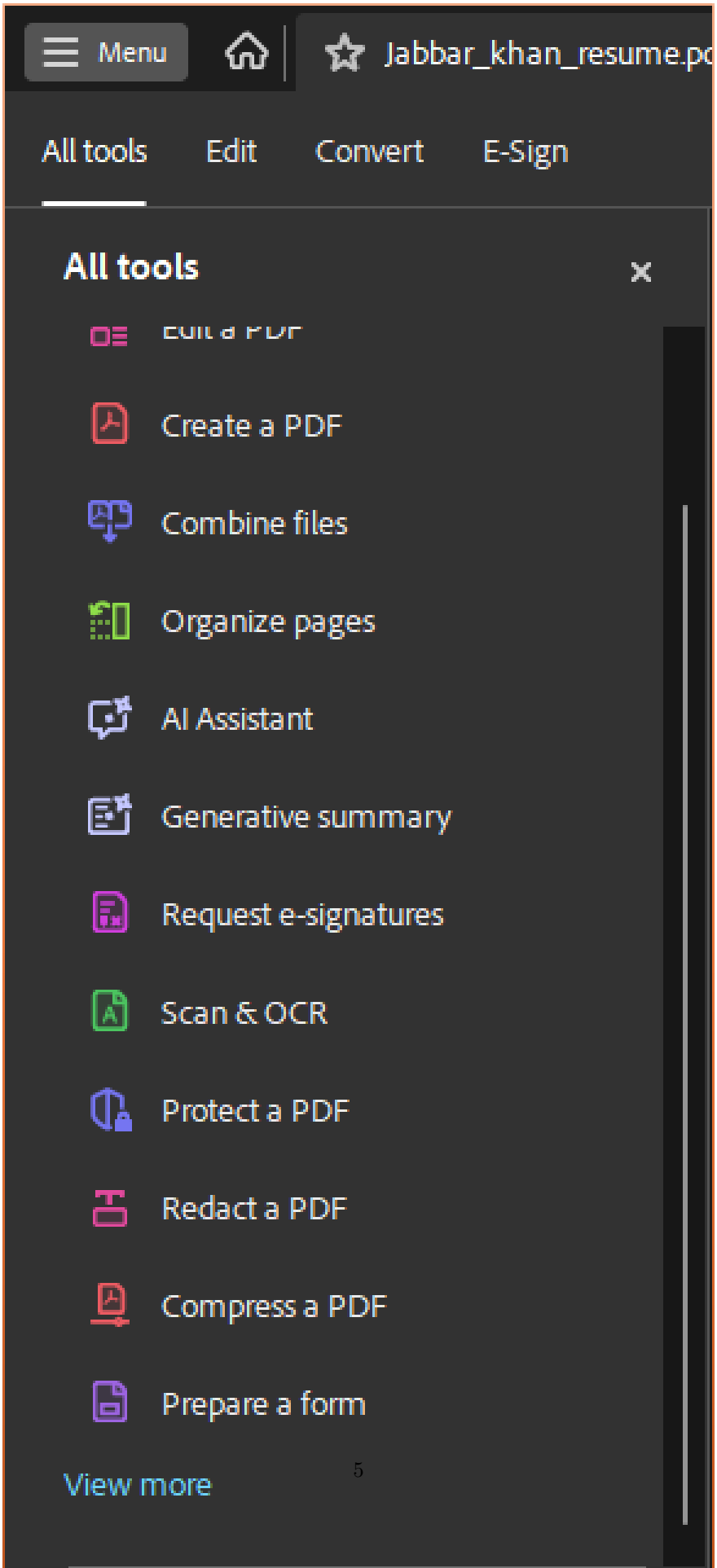


Figure 4: The main screen of the document viewer program.


Menu
Jabbar_khan_resume.p
All tools
Edit
Convert
E-Sign
All tools
Request e-signatures
Scan & OCR
Protect a PDF
Redact a PDF
Compress a PDF
Prepare a form
Fill & Sign
Add comments
Convert to PDF
Add a stamp
Use a certificate
Use print production


Menu
Jabbar_khan_resume.pd
All tools
Edit
Convert
E-Sign
All tools
Edit a PDF
Create a PDF
Combine files
Organize pages
AI Assistant
Generative summary
Request e-signatures
Scan & OCR
Protect a PDF
Redact a PDF
Compress a PDF
Prepare a form
View more

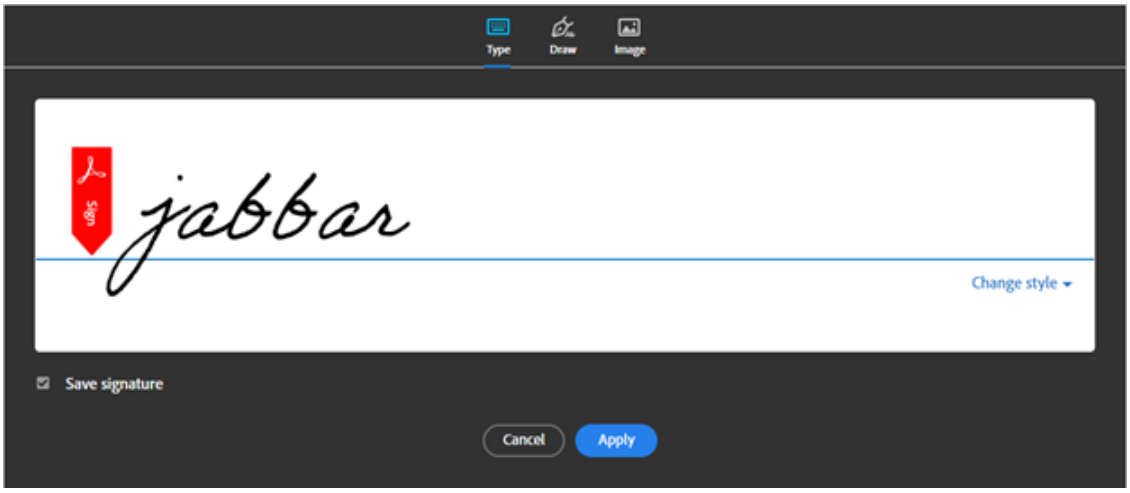


Figure 7: The window used to create the signature by typing a name.

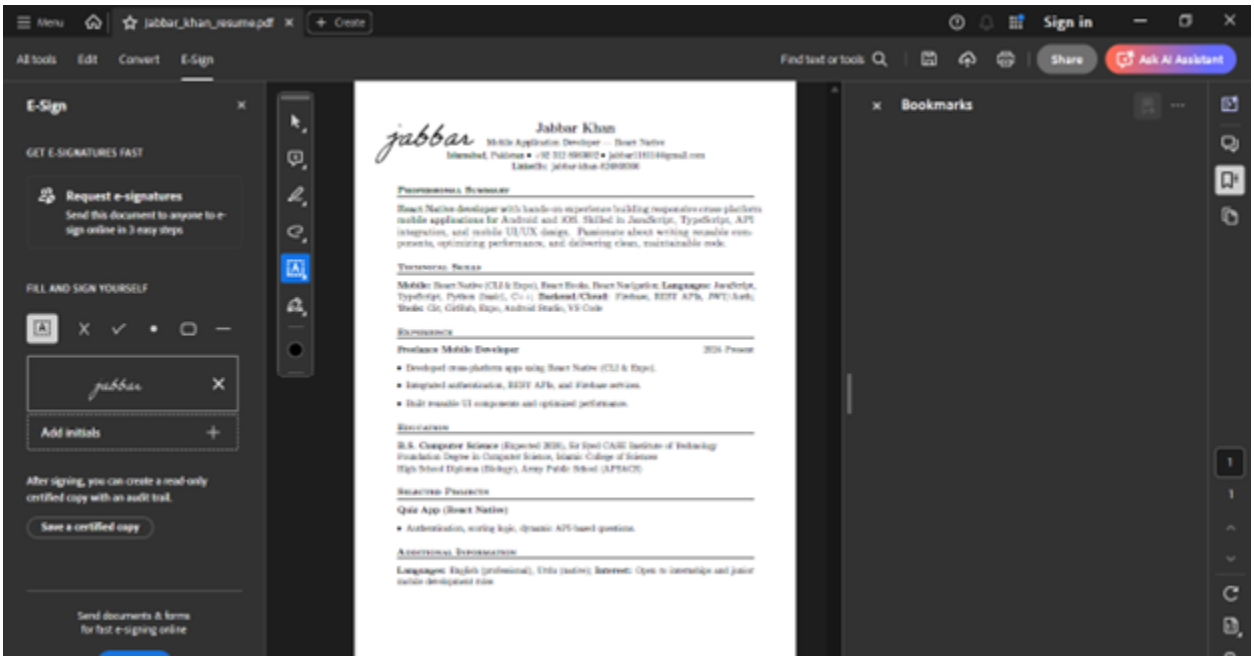


Figure 8: The final document with the signature successfully placed.