```
import React, { InputHTMLAttributes, ChangeEvent } from 'react';
import cn from 'classnames';

import s from './Input.module.css';

interface Props extends Omit<InputHTMLAttributes<any>, 'onChange'> {
  className?: string;
  onChange?: (value: string | ChangeEvent<HTMLInputElement> | any) => void;
  isEvent?: boolean;
}
const Input = (props: Props) => {
  const { className, children, isEvent, onChange, ...rest } = props;

  const rootClassName = cn(s.root, {}, className);

  const handleOnChange = (e: ChangeEvent<HTMLInputElement>) => {
    if (onChange) {
      if (isEvent) {
        onChange(e);
      } else {
        onChange(e.target.value);
      }
    }
    return null;
  };
  return (
    <input
      className={rootClassName}
      onChange={handleOnChange}
      autoComplete="off"
      autoCorrect="off"
      autoCapitalize="off"
      spellCheck="false"
      {...rest}
    />
  );
};

export default Input;
```